DC302 - smar First in Fieldbus

MANUAL DE INSTRUÇÕES
OPERAÇÃO E MANUTENÇÃO

FIELDBUS
E/S REMOTAS

AGO / 13

DC302

VERSÃO 3

FOUNDATION™

DC302MP

smar

www.smar.com.br

Especificações e informações estão sujeitas a modificações sem prévia consulta.
Informações atualizadas dos endereços estão disponíveis em nosso site.

web: www.smar.com/brasil2/faleconosco.asp

INTRODUÇÃO

Os equipamentos de campo disponíveis no sistema Fieldbus FOUNDATION™ não permitiam um sistema híbrido que aceitasse os sinais fieldbus e convencionais de Entrada/Saída. Um sistema misto é necessário durante a transição para a tecnologia Fieldbus. O DC302 permite uma integração fácil entre o Fieldbus e as Entradas/Saídas Convencionais.
Os equipamentos discretos como chaves de pressão, botoeiras, válvulas On/Off, bombas e esteiras podem ser integradas ao sistema FOUNDATION™ via barramento H1, usando o DC302. A alimentação, o controle e as E/S integradas em um mesmo equipamento, tornam o DC302 um equipamento compacto de fácil uso requerendo menos espaço em painéis quando comparados com outras soluções. O DC302 é parte integrante do SYSTEM302 e pode ser facilmente integrado em outros sistemas suportando o Fieldbus FOUNDATION™.
O DC302 permite que as entradas e saídas discretas e analógicas convencionais possam estar disponíveis e facilitar a configuração das estratégias de controle. Usando o conceito de Blocos Funcionais FOUNDATION™ e tornando o sistema homogêneo de modo a fazer com que estes equipamentos possam parecer como simples equipamentos em um barramento Fieldbus.
As malhas de controle são implementadas não importando se são equipamentos de E/S convencionais ou Fieldbus, necessitando apenas de uma linguagem de programação.
O DC302 é um equipamento de baixo custo que pode ser montado em trilho DIN, e que inclui a alimentação, o controle e os sinais de barramento H1, tudo em uma mesma unidade, tornando-o compacto e requerendo menos espaço em painéis, ao contrário de outras soluções.
Uma extensa biblioteca de Blocos Funcionais habilita o DC302 a executar a lógica e as funções de controle regulatório em sistemas integrados via barramento H1. Os blocos funcionais instanciáveis fornecem grande flexibilidade em estratégias de controle. As E/S discretas convencionais trabalham junto com os equipamentos puramente Fieldbus integrados numa mesma rede e numa mesma malha de controle. O DC302 é totalmente configurado pelo Syscon no SYSTEM302 ou por qualquer outra ferramenta de configuração Fieldbus. Os Blocos Funcionais fornecem lógicas como AND, OR, NAND etc, assim como, Flip-Flops, Timers e Contadores. A capacidade do “Link master” permite que o DC302 trabalhe como “backup LAS”, dando maior flexibilidade nas redes de comunicação.
O DC302 pode ser instalado perto de sensores e atuadores, eliminando a necessidade de extensos cabeamentos associados aos painéis e bandejas para as E/S convencionais, com subseqüente redução do custo do sistema. O DC302 torna possível que as conexões das E/S sejam distribuídas em várias localizações, e é ideal para conectar centros de controle de motores, equipamentos de velocidade variável, atuadores elétricos e válvulas operadas a motores pelo barramento Fieldbus H1.

Obtenha melhores resultados com o DC302 lendo cuidadosamente estas instruções.

ATENÇÃO
Este manual é compatível com a versão 3.XX, onde 3 denota a versão de software e XX indica o "release". Portanto, este manual é compatível com todos os "releases" da versão 3.

Exclusão de responsabilidade

O conteúdo deste manual está de acordo com o hardware e software utilizados na versão atual do equipamento. Eventualmente podem ocorrer divergências entre este manual e o equipamento. As informações deste documento são revistas periodicamente e as correções necessárias ou identificadas serão incluídas nas edições seguintes. Agradecemos sugestões de melhorias.

Advertência

Para manter a objetividade e clareza, este manual não contém todas as informações detalhadas sobre o produto e, além disso, ele não cobre todos os casos possíveis de montagem, operação ou manutenção.

Antes de instalar e utilizar o equipamento, é necessário verificar se o modelo do equipamento adquirido realmente cumpre os requisitos técnicos e de segurança de acordo com a aplicação. Esta verificação é responsabilidade do usuário.

Se desejar mais informações ou se surgirem problemas específicos que não foram detalhados e ou tratados neste manual, o usuário deve obter as informações necessárias do fabricante Smar. Além disso, o usuário está ciente que o conteúdo do manual não altera, de forma alguma, acordo, confirmação ou relação judicial do passado ou do presente e nem faz parte dos mesmos.

Todas as obrigações da Smar são resultantes do respectivo contrato de compra firmado entre as partes, o qual contém o termo de garantia completo e de validade única. As cláusulas contratuais relativas à garantia não são nem limitadas nem ampliadas em razão das informações técnicas apresentadas no manual.

Só é permitida a participação de pessoal qualificado para as atividades de montagem, conexão elétrica, colocação em funcionamento e manutenção do equipamento. Entende-se por pessoal qualificado os profissionais familiarizados com a montagem, conexão elétrica, colocação em funcionamento e operação do equipamento ou outro aparelho similar e que dispõem das qualificações necessárias para suas atividades. A Smar possui treinamentos específicos para formação e qualificação de tais profissionais. Adicionalmente, devem ser obedecidos os procedimentos de segurança apropriados para a montagem e operação de instalações elétricas de acordo com as normas de cada país em questão, assim como os decretos e diretivas sobre áreas classificadas, como segurança intrínseca, prova de explosão, segurança aumentada, sistemas instrumentados de segurança entre outros.

O usuário é responsável pelo manuseio incorreto e/ou inadequado de equipamentos operados com pressão pneumática ou hidráulica, ou ainda submetidos a produtos corrosivos, agressivos ou combustíveis, uma vez que sua utilização pode causar ferimentos corporais graves e/ou danos materiais.

O equipamento de campo que é referido neste manual, quando adquirido com certificado para áreas classificadas ou perigosas, perde sua certificação quando tem suas partes trocadas ou intercambiadas sem passar por testes funcionais e de aprovação pela Smar ou assistências técnicas autorizadas da Smar, que são as entidades jurídicas competentes para atestar que o equipamento como um todo, atende as normas e diretivas aplicáveis. O mesmo acontece ao se converter um equipamento de um protocolo de comunicação para outro. Neste caso, é necessário o envio do equipamento para a Smar ou à sua assistência autorizada. Além disso, os certificados são distintos e é responsabilidade do usuário sua correta utilização.

Respeite sempre as instruções fornecidas neste Manual. A Smar não se responsabiliza por quaisquer perdas e/ou danos resultantes da utilização inadequada de seus equipamentos. É responsabilidade do usuário conhecer as normas aplicáveis e práticas seguras em seu país.

ÍNDICE

SEÇÃO 1 - INSTALAÇÃO 1.1
GERAL 1.1
MONTAGEM 1.1
CONEXÃO ELÉTRICA 1.3
TOPOLOGIA E CONFIGURAÇÃO DA REDE 1.5
SISTEMA GERAL 1.6

SEÇÃO 2 - OPERAÇÃO 2.1
DESCRIÇÃO FUNCIONAL – ELETRÔNICA 2.1
UNIDADE CENTRAL DE PROCESSAMENTO (CPU), RAM , FLASH, EEPROM 2.1
CONTROLADOR DA COMUNICAÇÃO 2.1
FONTE DE ALIMENTAÇÃO 2.1
INICIALIZAÇÃO DE FÁBRICA (FACTORY RESET) 2.1
LATCHES DE ENTRADA 2.1
LATCHES DE SAÍDA 2.1
ISOLAÇÃO ÓTICA 2.2

SEÇÃO 3 - CONFIGURAÇÃO 3.1
CONEXÃO FÍSICA AO BLOCO DI (ENTRADA DIGITAL) 3.1
CONEXÃO FÍSICA AO BLOCO DO (SAÍDA DIGITAL) 3.2
CONEXÃO FÍSICA AO BLOCO MDI (MÚLTIPLAS ENTRADAS DIGITAIS) 3.2
CONEXÃO FÍSICA AO BLOCO MDO (MÚLTIPLAS SAÍDAS DIGITAIS) 3.3
CONEXÃO FÍSICA AO BLOCO PID STEP 3.3
JUMPER DE SIMULAÇÃO (SIMULATION JUMPER) 34
EXEMPLOS DE APLICAÇÕES 3.5
BLOCO FUNCIONAL FLEXÍVEL 3.7
DESCRIÇÃO 3.7
BLOCK_ERR 3.7
STATUS 3.8
MODOS SUPORTADOS 3.8
PARÂMETROS 3.8
FUNÇÕES 3.15
TP TIMER PULSE 3.15
TON TIMER ON-DELAY 3.16
TOF TIMER OFF-DELAY 317
CTD PULSE COUNTER DOWN 317
CTU PULSE COUNTER UP 318
FLIP-FLOP RS 3.18
FLIP-FLOP SR 3.18
CÓDIGOS DE ERROS 3.19
EXEMPLOS DE APLICAÇÕES 3.20
PROCESSO: 3.22

SEÇÃO 4 - PROCEDIMENTOS DE MANUTENÇÃO 4.1
GERAL 4.1
PROCEDIMENTO DE DESMONTAGEM 4.1
PROCEDIMENTO DE MONTAGEM 4.2
PROCEDIMENTOS DE ATUALIZAÇÃO DO FIRMWARE DO DC302 ..... 4.2
ACESSÓRIOS 4.2
PARTES SOBRESSALENTES 4.3

SEÇÃO 5 - CARACTERÍSTICAS TÉCNICAS 5.1
GERAL 5.1
ENTRADAS DO DC302 ..... 5.2
DESCRIÇÃO DAS ENTRADAS 5.2
ESPECIFICAÇÕES TÉCNICAS 5.2
DC302: SAÍDAS EM COLETOR ABERTO 5.3
DESCRIÇÃO - SAÍDAS 5.3

ESPECIFICAÇÕES TÉCNICAS 5.3
CÓDIGO DE PEDIDO 5.3

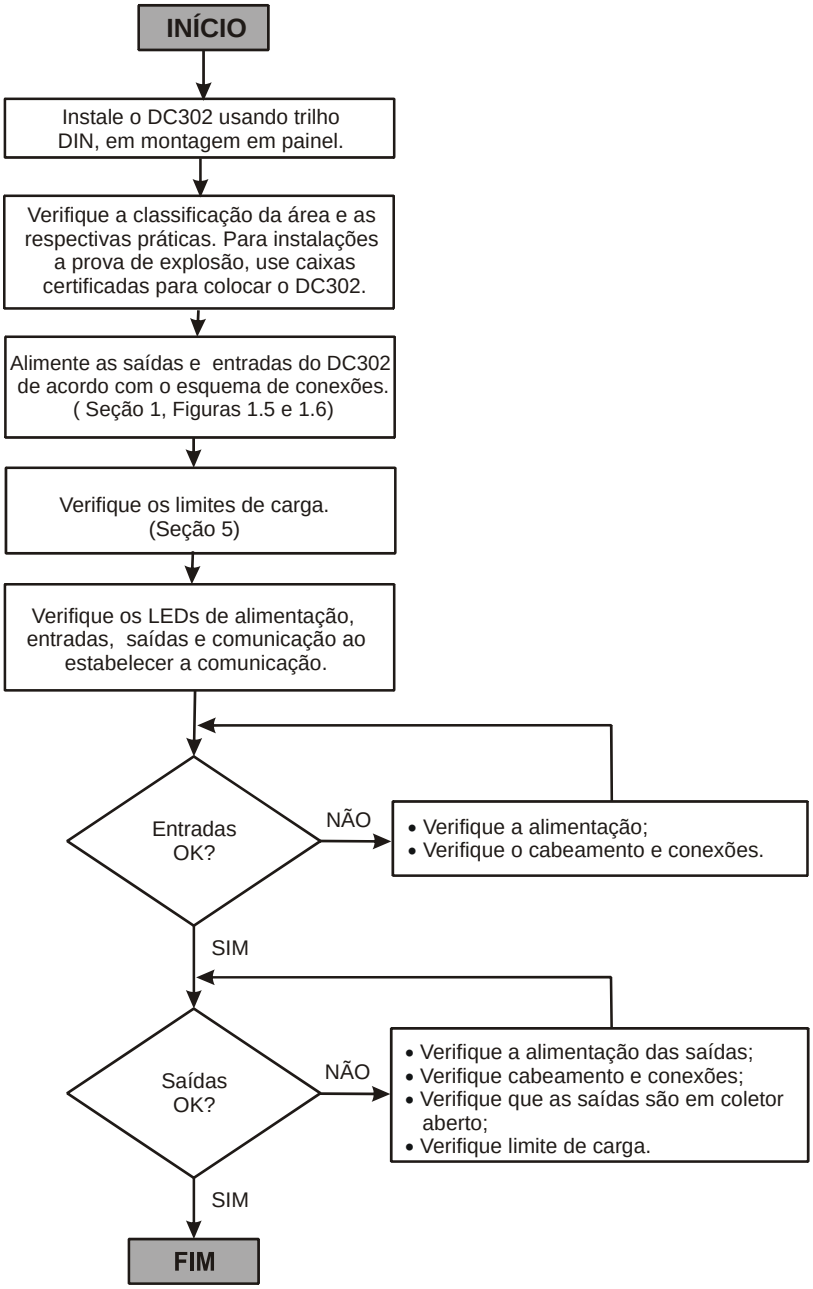
APÊNDICE A – FSR – FORMULÁRIO PARA SOLICITAÇÃO DE REVISÃO A.1
RETORNO DE MATERIAIS A.2

APÊNDICE B – TERMO DE GARANTIA SMAR B.1

Fluxograma de Instalação

(Guia Rápido)

ATENÇÃO
Obtenha melhores resultados do DC302 lendo cuidadosamente o manual completo.


INÍCIO
Instale o DC302 usando trilho DIN, em montagem em painel.
Verifique a classificação da área e as respectivas práticas. Para instalações a prova de explosão, use caixas certificadas para colocar o DC302.
Alimente as saídas e entradas do DC302 de acordo com o esquema de conexões. (Seção 1, Figuras 1.5 e 1.6)
Verifique os limites de carga. (Seção 5)
Verifique os LEDs de alimentação, entradas, saídas e comunicação ao estabelecer a comunicação.
Entradas OK?
NÃO
• Verifique a alimentação;
• Verifique o cabeamento e conexões.
SIM
Saídas OK?
NÃO
• Verifique a alimentação das saídas;
• Verifique cabeamento e conexões;
• Verifique que as saídas são em coletor aberto;
• Verifique limite de carga.
SIM
FIM

* Maiores informações encontram-se na Seção 1 do manual de instalação, configuração e manutenção do DC302.

INSTALAÇÃO

Geral

A precisão de uma medição em controle depende de várias variáveis. Embora o DC302 tenha um alto desempenho de performance, uma instalação adequada é necessária para se aproveitar ao máximo os benefícios oferecidos.

De todos os fatores que podem afetar a precisão, as condições ambientais são as mais difíceis de controlar. Entretanto, há maneiras de se reduzir os efeitos de temperatura, umidade e vibração.

Localizando o DC302 em áreas protegidas de mudanças bruscas ambientais, pode-se melhorar sua performance.
Em ambientes quentes, o DC302 deve ser instalado de forma a se evitar ao máximo a exposição a raios solares. Deve-se evitar a instalação próxima a linhas ou vasos com altas temperaturas.

A umidade é fatal aos circuitos eletrônicos. Em áreas com alto índice de umidade relativa deve-se certificar da correta instalação e proteção.

Para detalhes de montagem refira às Figuras 1.1 e 1.2.

Montagem

Use o trilho DIN (TS35-DIN EN 50022 ou TS32-DIN EN50035 ou TS15 DIN EN50045), como é mostrado na Figura 1.1 – Montagem Mecânica. O DC302 pode ser opcionalmente fornecido em uma caixa de distribuição a prova de explosão.

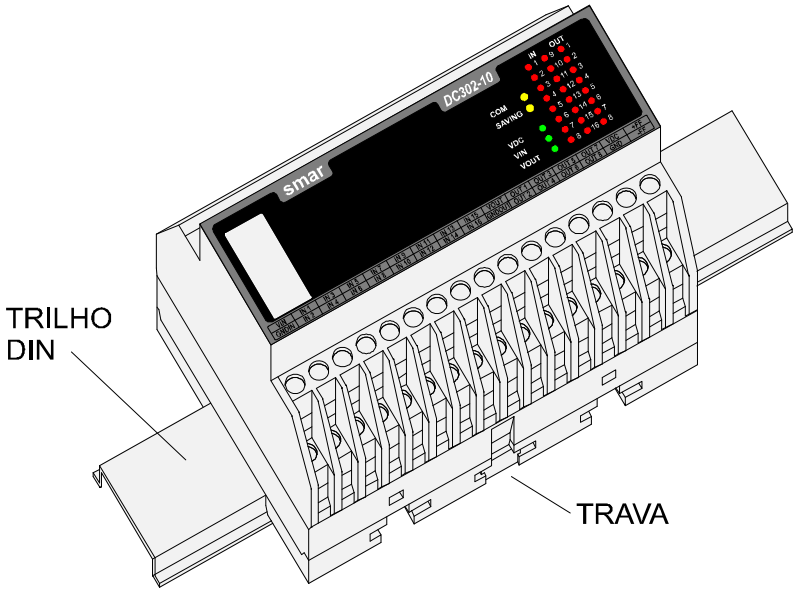
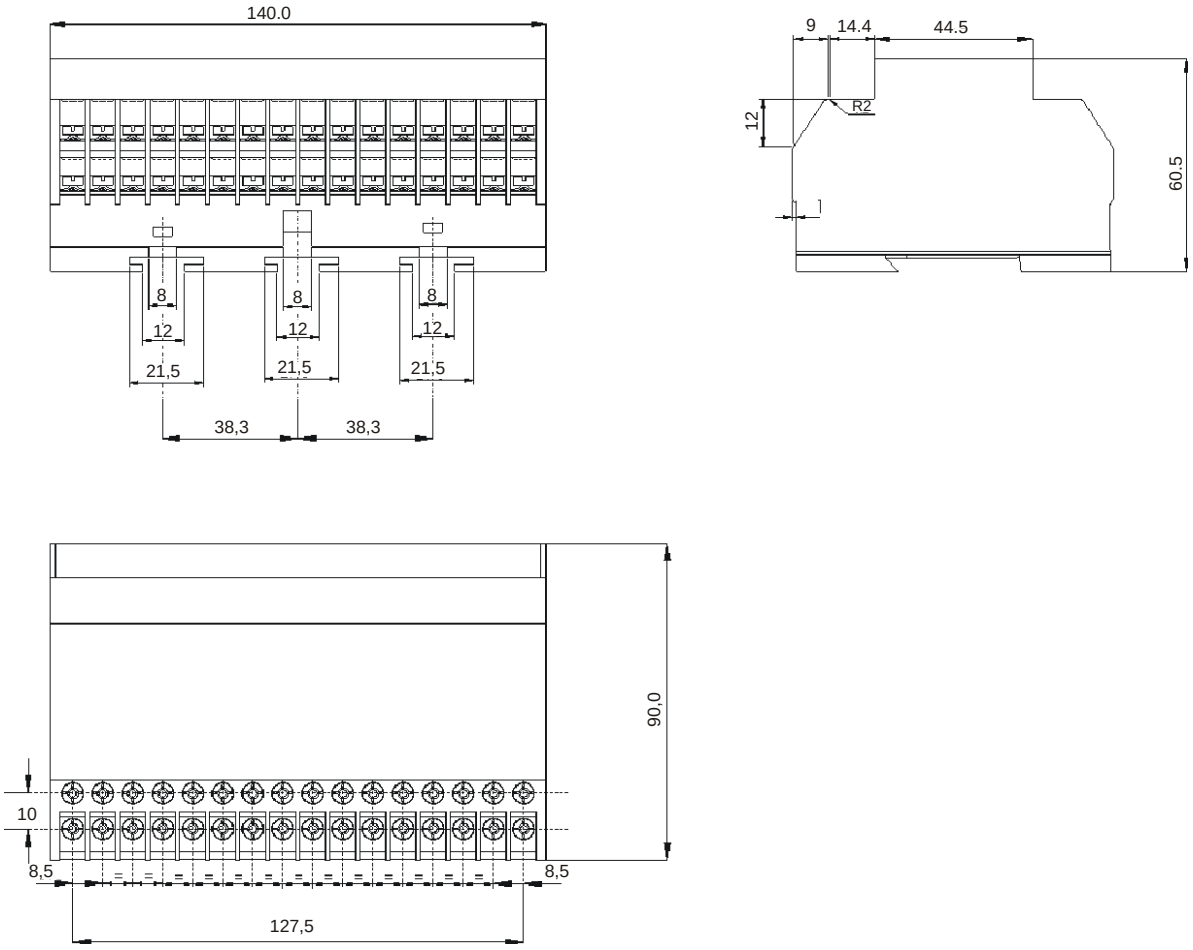


Figura 1.1- Montagem Mecânica



NOTA
As medidas são em mm.

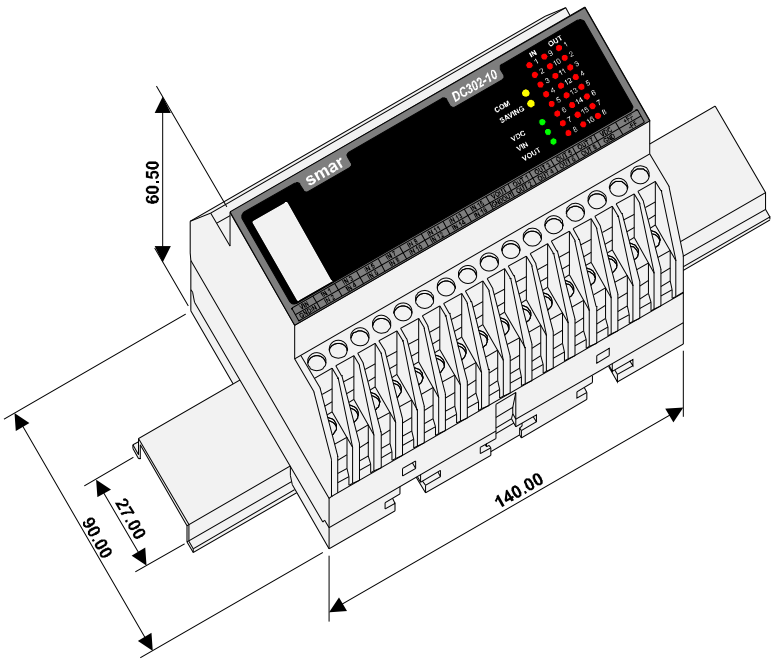


Figura 1.2 – Montagem Mecânica e desenho dimensional do DC302

Conexão Elétrica

Acesse o conjunto de conexão na vista frontal através da etiqueta de conexão para as entradas, saídas e fonte de alimentação. As conexões são feitas através dos parafusos.

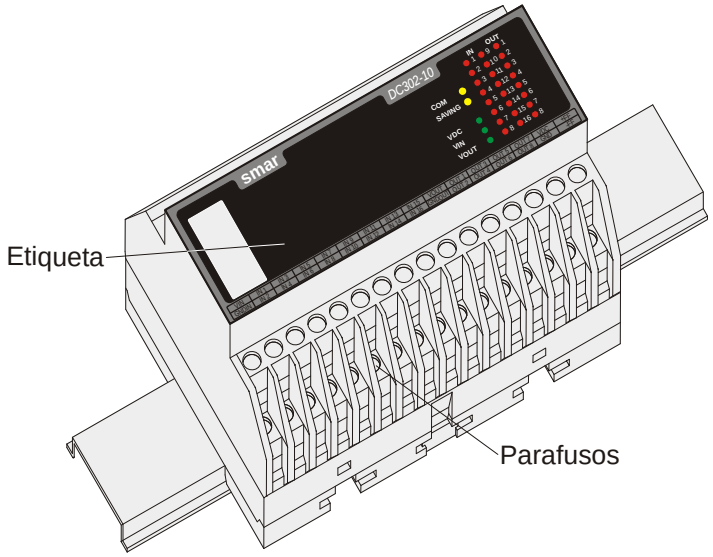


Figura 1.3 – Bloco de conexões

A tabela a seguir descreve os terminais do DC302:

Terminal	Superiot (S)	Inferior (I)	Comentário
1	VIN (1S)	GNDIN (1I)	Alimentação auxiliar para as entradas.
2	IN1 (2S)	IN2 (2I)	Entradas digitais.
3	IN3 (3S)	IN4 (3I)	
4	IN5 (4S)	IN6 (4I)	
5	IN7 (5S)	IN8 (5I)	
6	IN9 (6S)	IN10 (6I)	
7	IN11 (7S)	IN12 (7I)	
8	IN13 (8S)	IN14 (8I)	
9	IN15 (9S)	IN16 (9I)	
10	VOUT (10S)	GNDOUT (10I)	Alimentação auxiliar para as saídas.
11	OUT1 (11S)	OUT2 (11I)	Saídas digitais.
12	OUT3 (12S)	OUT4 (12I)	
13	OUT5 (13S)	OUT6 (13I)	
14	OUT7 (14S)	OUT8 (14I)	
15	VDC (15S)	GND (15I)	Alimentação auxiliar para a comunicação.
16	+FF (16S)	-FF (16I)	Sinal de comunicação FOUNDATION fieldbus™

Tabela1.1 – Bloco de conexões

As conexões utilizadas devem ser feitas de acordo com a aplicação. Por exemplo, refira as figuras 1.4 e 1.5.

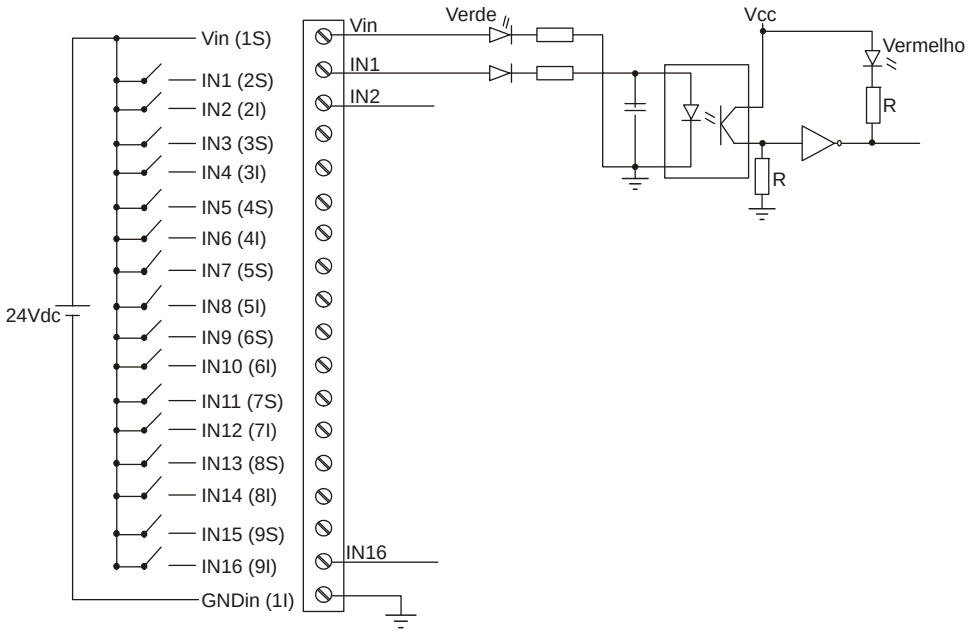


Figura 1.4 – Exemplo de conexões de entrada

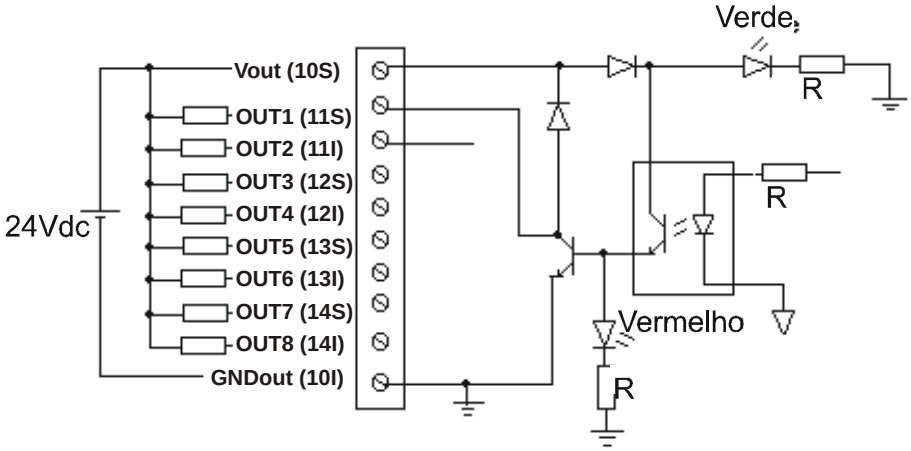


Figura 1.5 – Exemplo de conexões de saída

O **DC302** não é um equipamento alimentado via barramento. O DC302 usa a taxa de 31,25 Kbit/s em modo de tensão para a sinalização física. Muitos tipos de equipamentos Fieldbus podem ser conectados no mesmo barramento, podendo ser alimentados por este ou não. Quando alimentados, devem usar a mesma sinalização. Até 16 equipamentos podem ser conectados em paralelo pelo mesmo para de fios.

O **DC302** não é alimentado pelo barramento. Quando não for requerida a especificação para segurança intrínseca, pode-se conectar até 16 equipamentos Fieldbus no barramento.

Em áreas perigosas, o número de equipamentos deve ser limitado às restrições de segurança intrínseca.

O **DC302** é protegido contra polaridade reversa e pode suportar até ±35 VDC sem danos.

NOTA
Favor referir ao Manual de instalação Fieldbus para maiores detalhes.

ATENÇÃO

ÁREAS PERIGOSAS

Em áreas perigosas que exigem segurança intrínseca ou cuidada em relação a explosões, as entidades de circuito e instalações devem ser observadas.

O acesso dos cabos de sinal aos terminais de ligação pode ser feito utilizando-se eletrodutos e conduítes.

Se outras certificações forem necessárias, refira-se ao certificado ou à norma específica para as restrições de instalação.

Topologia e Configuração da rede

A topologia em Barramento (Veja Figura 1.6 – Topologia Barramento) e topologia em Árvore (Veja Figura 1.7 – Topologia Árvore) são suportadas. Ambos os tipos possuem um barramento principal com dois terminadores. Os equipamentos são conectados ao tronco principal através das derivações (braços). As derivações podem ser integradas aos equipamentos de tal forma a resultar um comprimento igual a zero.

Em uma derivação podem ser conectados mais de um equipamento, dependendo do comprimento da mesma. Acopladores ativos podem ser usados para se estender o comprimento da derivação.

O comprimento total do cabeamento, incluindo as derivações entre dois equipamentos não deve exceder a 1900m.

A conexão das caixas de junções deve ser mantida em até 15 para cada 250m.

ATENÇÃO

Fontes de Alimentação

Onde se requer isolação entre as fontes das entradas e saídas, recomenda-se usar no mínimo duas fontes de alimentação, sendo uma para as entradas e outra para as saídas e VDC.

Onde a aplicação não exigir isolação entre as fontes, pode-se ter somente uma fonte para entradas, saídas e VDC.

As saídas e entradas são isoladas opticamente.

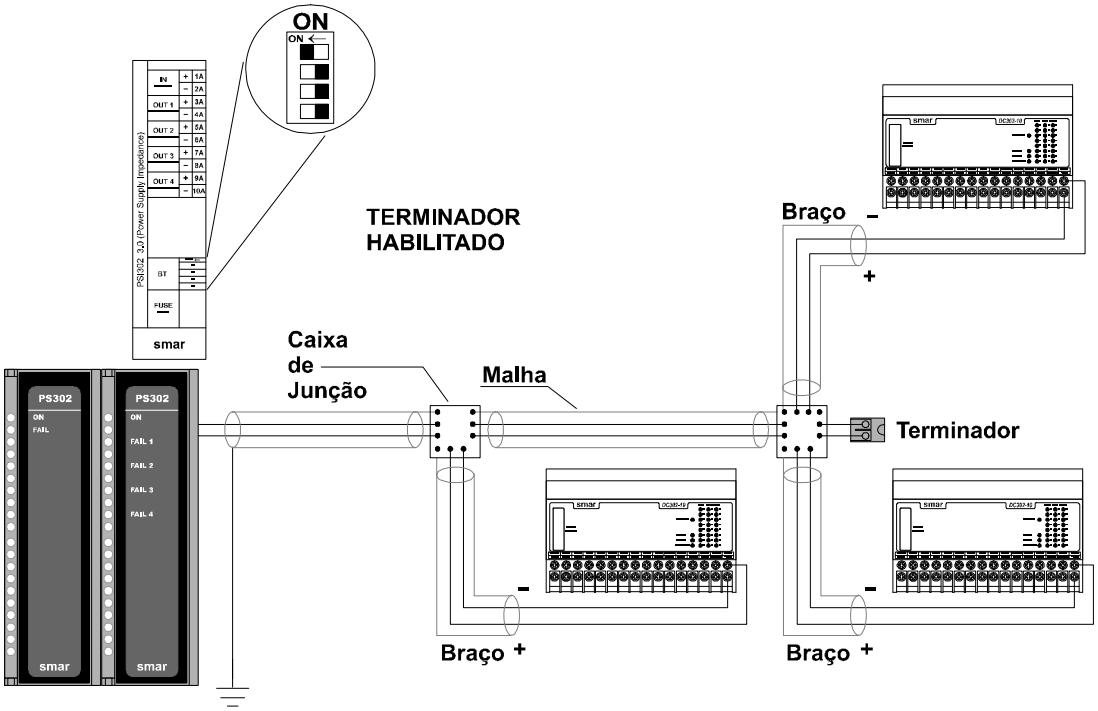


Figura 1.6- Topologia Barramento

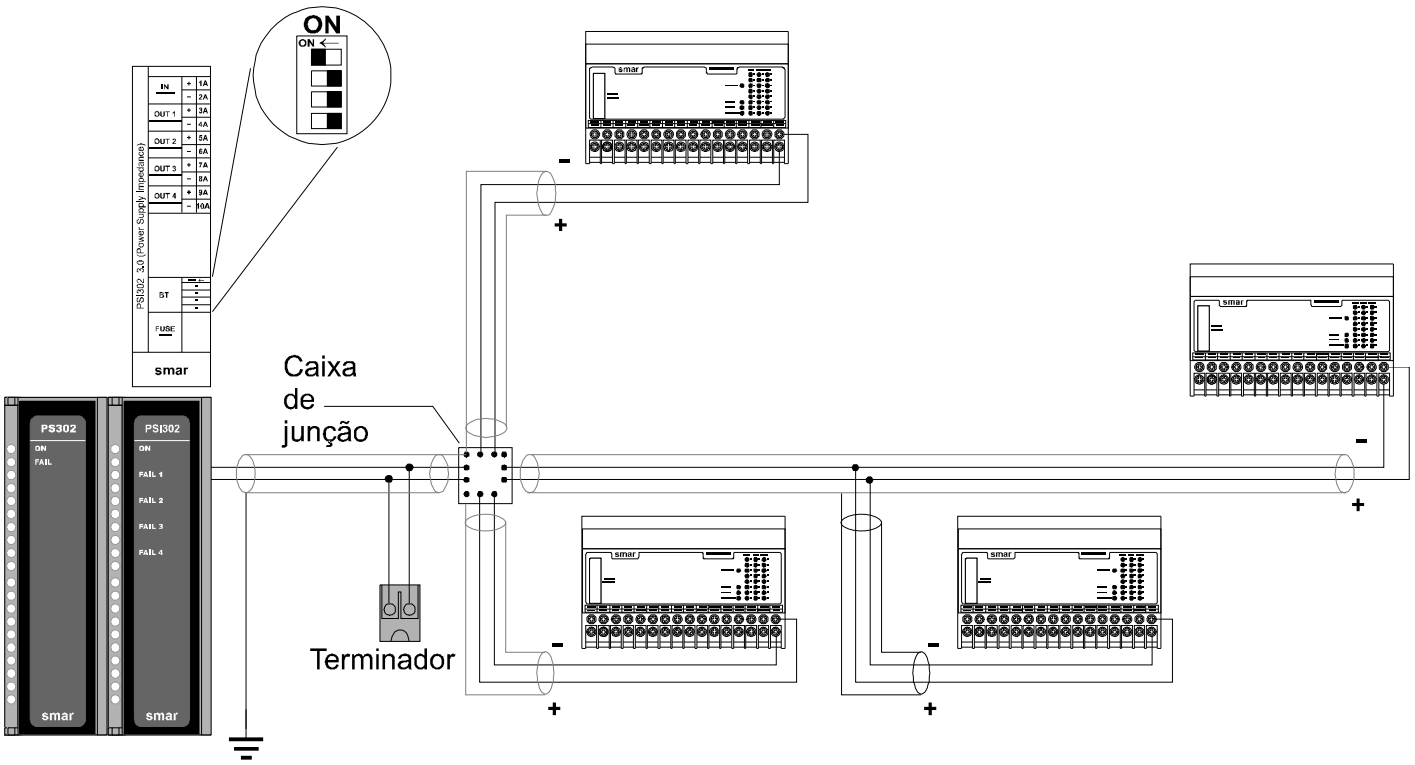


Figura 1.7 – Topologia Árvore

Sistema Geral

De acordo com a figura a seguir, veja uma topologia de rede genérica onde o DC302 é integrado em uma rede Fieldbus simples.

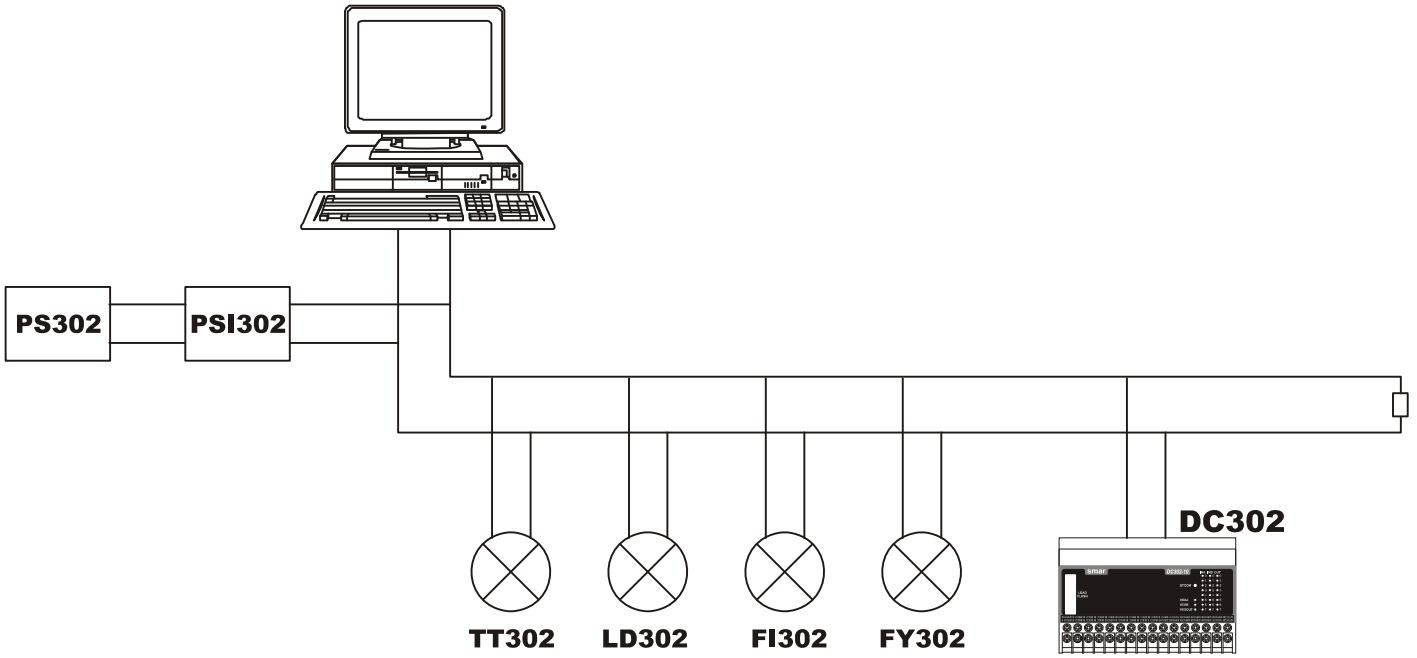


Figura 1.8 – DC302 e um Sistema Genérico Fieldbus

OPERAÇÃO

O **DC302** recebe até 16 entradas isoladas oticamente e pode acionar até oito saídas em coletor aberto, de tal forma a interfacear pontos discretos ao Sistema Fieldbus.

Uma extensa biblioteca de Blocos Funcionais habilita o DC302 a executar a lógica e funções de controle regulatório e discreta integrada via barramento H1. Blocos funcionais instanciáveis fornecem grande flexibilidade em estratégias de controle.

As E/S discretas convencionais trabalham junto com os equipamentos puramente Fieldbus integrados numa mesma rede e numa mesma malha de controle.
Blocos Funcionais de Saídas incluem procedimentos padrões de mecanismo de segurança em caso de falhas segundo o Fieldbus FOUNDATION™.

Entradas e saídas são isoladas umas das outras e acessadas via rede de comunicação através dos canais dos blocos funcionais. Os LEDs são utilizados para indicar o estado das entradas e saídas. O uso dos Blocos Funcionais FOUNDATION™ torna o sistema homogêneo de tal forma que equipamento de entradas e saídas discretas e analógicas convencionais possam estar disponíveis para facilitar a configuração de estratégias de controle, parecendo como simples equipamentos em um barramento fieldbus.

Descrição Funcional – Eletrônica

Veja o diagrama de blocos (Figura 2.1 – *Diagrama de Blocos DC302*). A função de cada bloco é descrita a seguir.

Unidade Central de Processamento (CPU), RAM , FLASH, EEPROM

A CPU é a parte inteligente do DC302, sendo responsável pelo gerenciamento e operação do bloco de execução, autodiagnose e comunicação. O programa é armazenado em uma memória Flash e os dados temporários em uma memória RAM. Na falta de energia os dados armazenados na RAM são perdidos. A memória EEPROM armazena os dados não-voláteis que serão usados posteriormente. Exemplos de tais dados são: calibração, configuração e dados de identificação.

Controlador da Comunicação

É responsável pela monitoração da atividade da linha, modulação e demodulação dos sinais do barramento.

Fonte de Alimentação

Alimenta os circuitos do DC302.

Inicialização de Fábrica (Factory Reset)

O DC302 possui uma área no lado superior esquerdo do seu envólucro onde pode ser visto a inscrição de factory reset. Para efetuar esta operação, basta um curto-circuito nos contatos na placa de circuito impresso, ligando o DC302 nesta condição de curto e mantendo a chave até o led de saving acenda.

Existem 2 contatos que permitem a inicialização de fábrica.

Latches de Entrada

São latches que armazenam as condições das entradas.

Latches de saída

São latches que armazenam as condições das saídas.

Isolação Ótica

Isolação Ótica para as entradas e saídas.

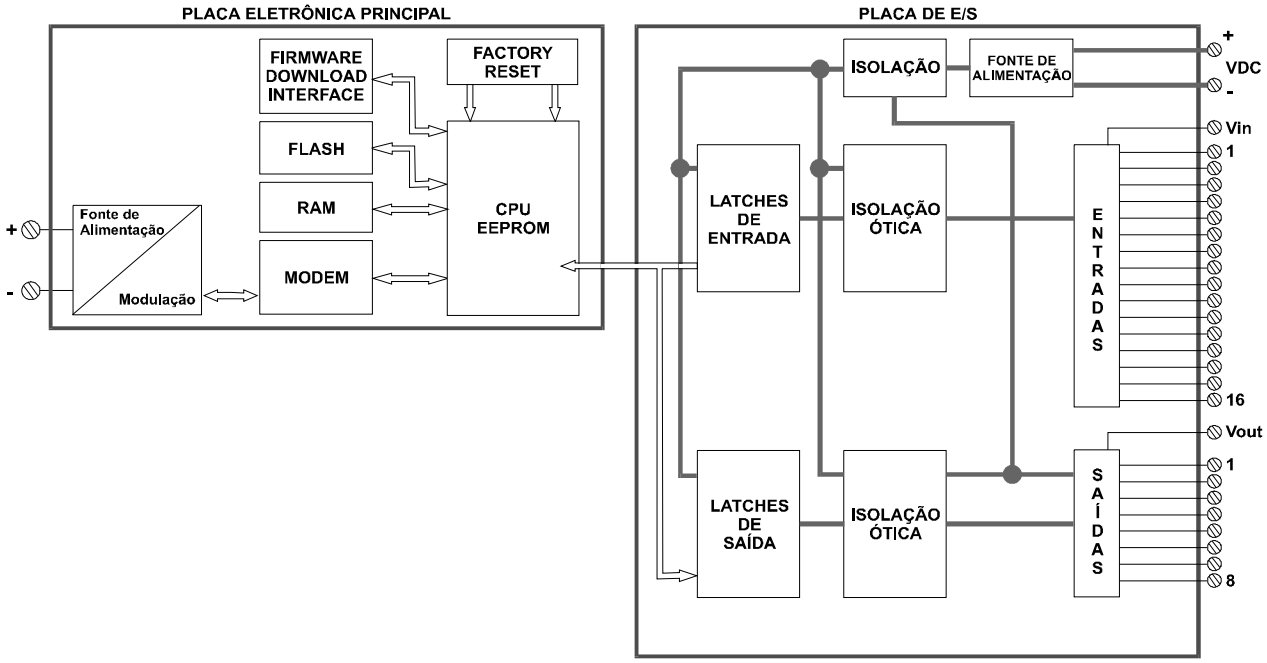


Figura 2.1 –Diagrama de Blocos DC302

CONFIGURAÇÃO

O **DC302** pode ser configurado via Syscon ou qualquer outra ferramenta segundo os padrões Fieldbus FOUNDATION™.

O **DC302** possui vários Blocos Funcionais como: Flip-Flop, Edge Trigger, Analog Alarm, Timer e Logic, Discrete Input, Discrete Output, Multiple Discrete Input, Multiple, Discrete Output, Arithmetic, Input Select, PID controller, PID Step e Flexible Function Block.

Os Blocos Funcionais não são citados neste manual. Para explicações e detalhes, refira-se ao manual de Blocos Funcionais.

O DC302 pode compartilhar seus blocos funcionais com outros equipamentos utilizando o SYSCON. Para usá-lo refira-se ao manual do SYSCON.

NOTA
A versão de firmware do equipamento deve ser utilizada com a versão de DD/CFF correspondente.

VERSÃO DE FIRMWARE	VERSÃO DE DD
3.46	0501
3.47	0501
3.48	0502
3.49	0502
3.50	0502
3.50A	0502
3.50C	0502
3.50F	0502
3.50J	0502
3.50R	0502
3.50S	0502
3.50T	0503
3.51	0801
3.52	0601
3.52F	0601
3.55	0801
3.55A	0801
3.56	0901

Conexão Física ao Bloco DI (Entrada Digital)

O Bloco DI utiliza um dado discreto de entrada, selecionado via canal e o deixa disponível para outro bloco funcional através de sua saída.
Para maiores informações e detalhes, refira-se ao manual dos Blocos Funcionais.

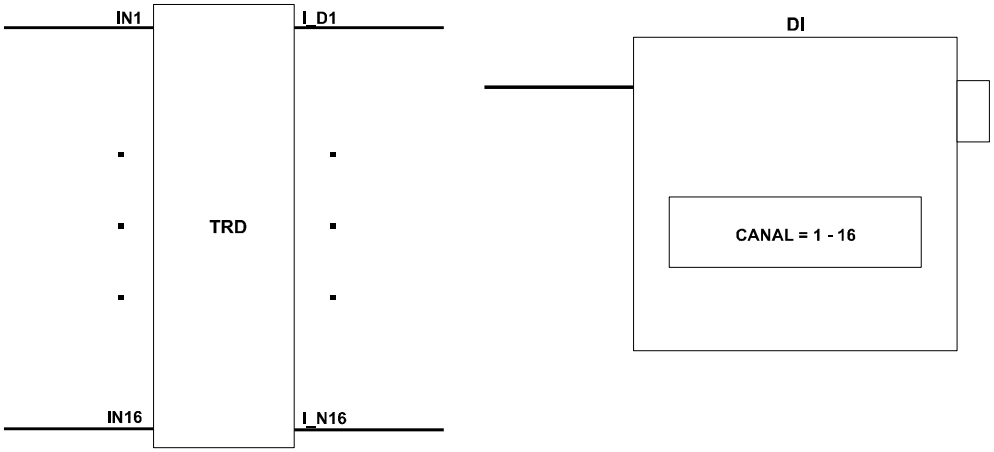


Figure 3.1 – O DC302 e as conexões com o Bloco DI.

Conexão Física ao Bloco DO (Saída Digital)

O bloco funcional DO converte o valor de SP_D para um valor útil ao hardware, através do canal selecionado.
Para maiores informações e detalhes, por favor, referencie-se ao manual de Blocos Funcionais.

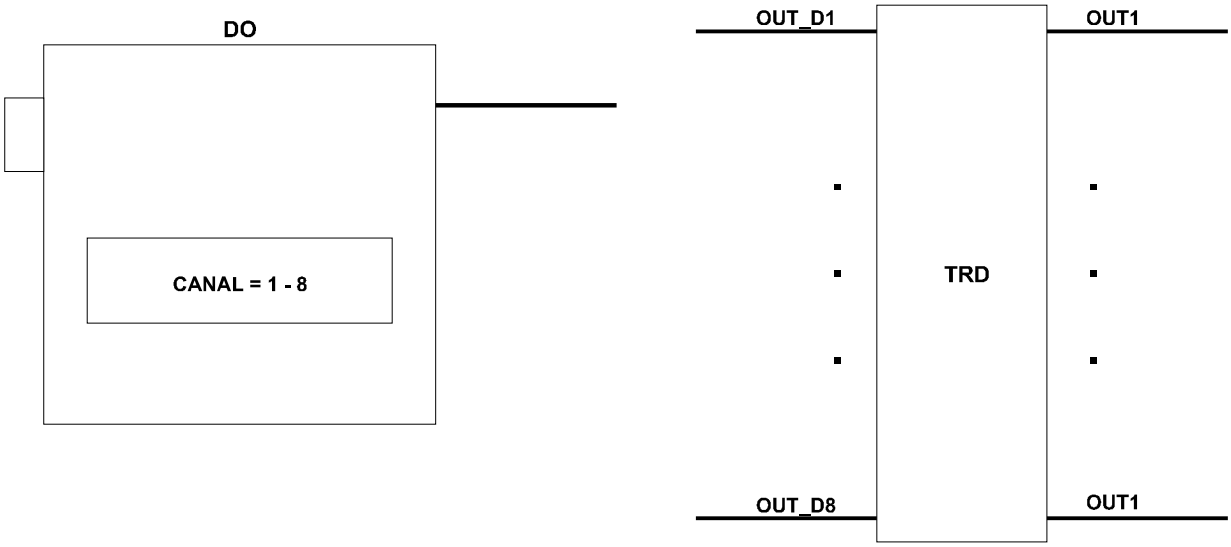


Figure 3.2 - O DC302 e as conexões com o Bloco DO.

Conexão Física ao Bloco MDI (Múltiplas Entradas Digitais)

Um MDI disponibiliza à rede Fieldbus 8 variáveis discretas das entradas físicas, através dos parâmetros OUT_D1 a OUT_D8. As condições de indicação dos estados das entradas dependem do sistema de E/S.Por exemplo, se existe uma falha em um sensor de entrada, esta será indicada no “status” do parâmetro OUT_Dx. Problemas na interface de E/S serão indicados como “BAD – Device Failure” (Sinal ruim – Falha no dispositivo).
Para maiores informações e detalhes, por favor, refira-se ao manual de Blocos Funcionais.

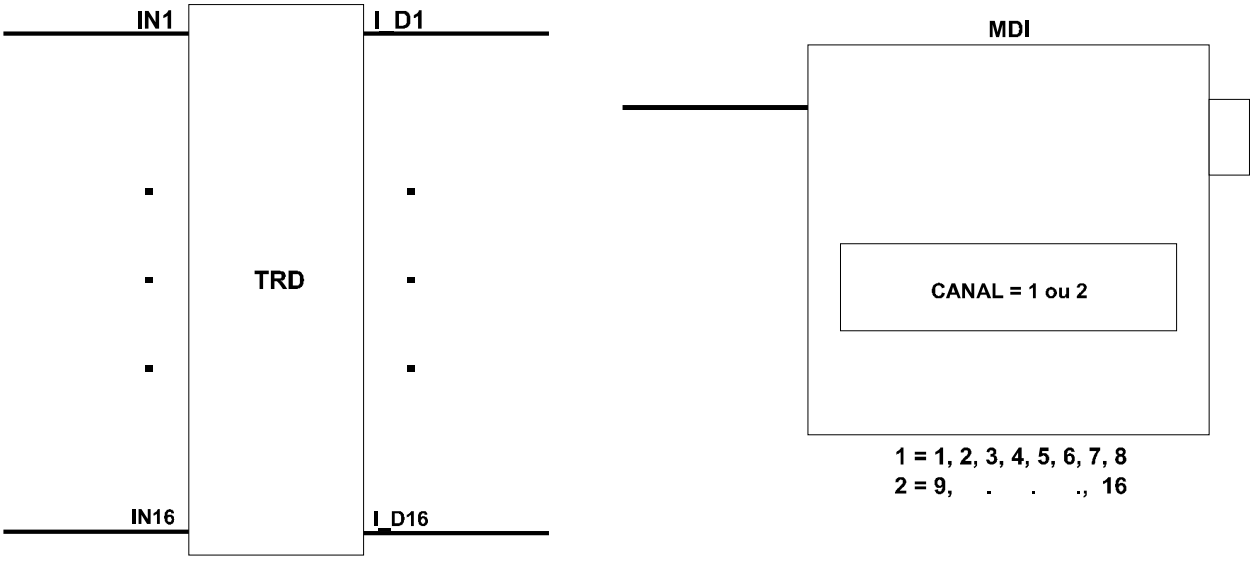


Figure 3.3 – O DC302 e as conexões com o Bloco MDI

Conexão Física ao Bloco MDO (Múltiplas Saídas Digitais)

Este bloco fornece uma maneira de enviar 8 variáveis discretas às saídas físicas através dos parâmetros IN_D1 a IN_D8.
Para maiores informações e detalhes, por favor, referencie-se ao manual de Blocos Funcionais.

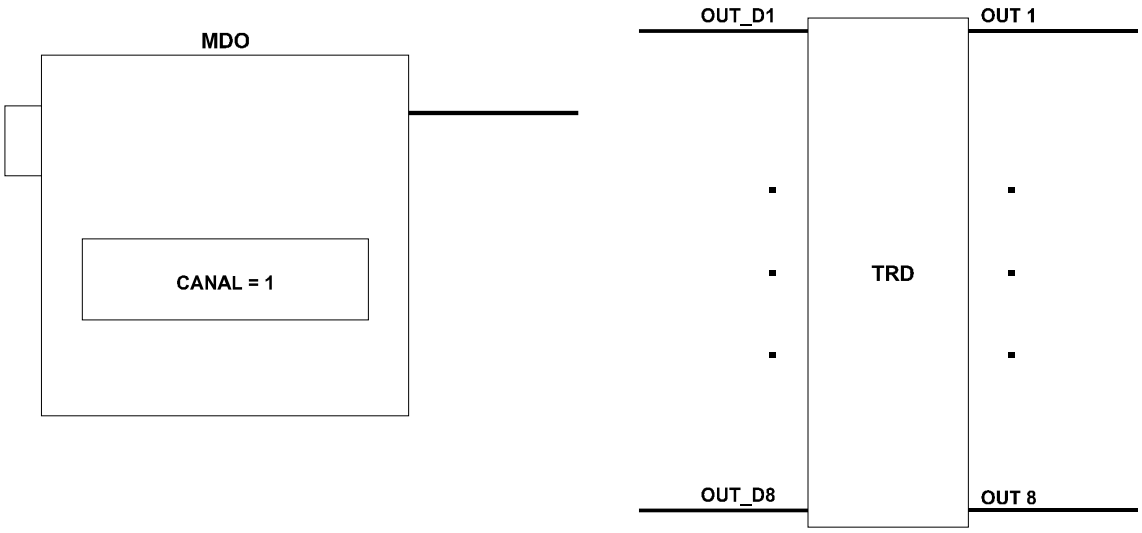


Figure 3.4 - O DC302 e as conexões com o Bloco MDO

Conexão Física ao Bloco PID Step

O Bloco Funcional PID Step é comumente utilizado quando o elemento final de controle tem um atuador acionado por um motor elétrico. O elemento final de controle é posicionado rotacionando-se o motor em sentido horário ou anti-horário com o acionamento discreto para cada direção. Em um controle de válvula, por exemplo, se faz necessário um sinal para abrir e outro para fechar. Se nenhum dos sinais está presente, a haste se mantém na posição atual. Para maiores informações e detalhes refira-se ao manual dos Blocos Funcionais.

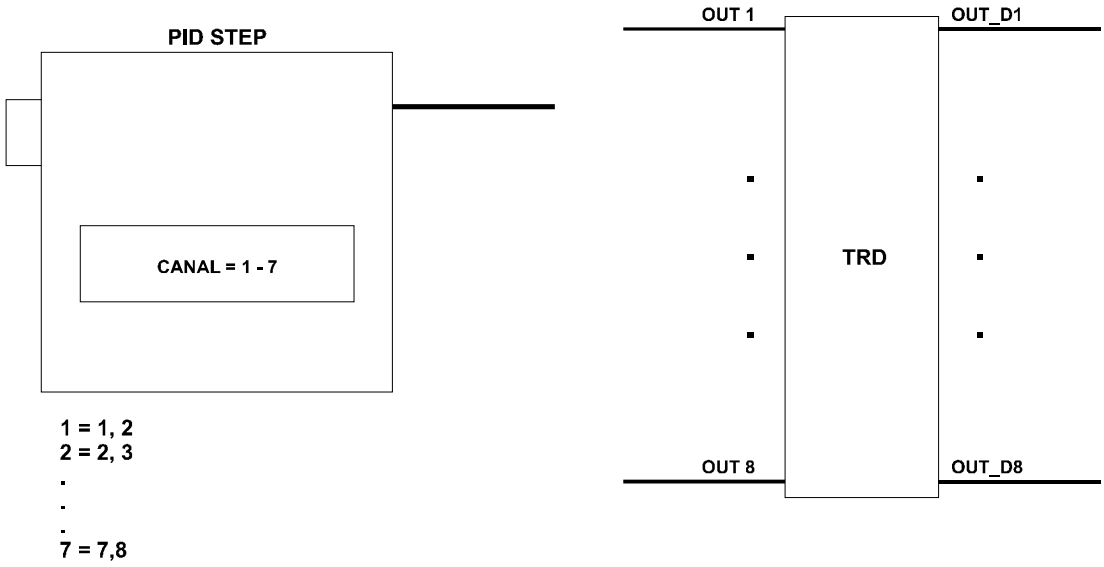


Figure 3.5 - O DC302 e as conexões com o Bloco PID Step

Jumper de Simulação (Simulation Jumper)

Para realizar a simulação dos blocos é necessário primeiro habilitar o *jumper simulation*, ele deve estar na posição ON no DC302. Veja habilitação do hardware na figura abaixo. Em seguida, habilite o bloco DI ou DO no software de configuração, refira-se aos manuais dos Blocos Funcionais: “Library A” e “Library B“.

O parâmetro SIMULATE é usado para os propósitos de diagnóstico e verificação. Quando está ativo, o valor e status do transdutor serão substituídos pelo valor simulado e status.

Vista superior do DC302 com detalhe do Simulation Jumper habilitado na posição ON.

NOTA
O parâmetro SIMULATE pode ser desabilitado por software no parâmetro SIMULATE ou no hardware, através do jumper na posição OFF. Refira-se aos manuais dos Blocos Funcionais: “Library A” e “Library B “para obter detalhes referente a configuração para simulação dos blocos. Veja os itens: • Simulação; • Bloco Funcional de Entrada: DI – Entrada Discreta; • Bloco Funcional de Saída: DO – Saída Discreta.

Exemplos de Aplicações

Aplicação 1: Um computador pode manipular as entradas e saídas.

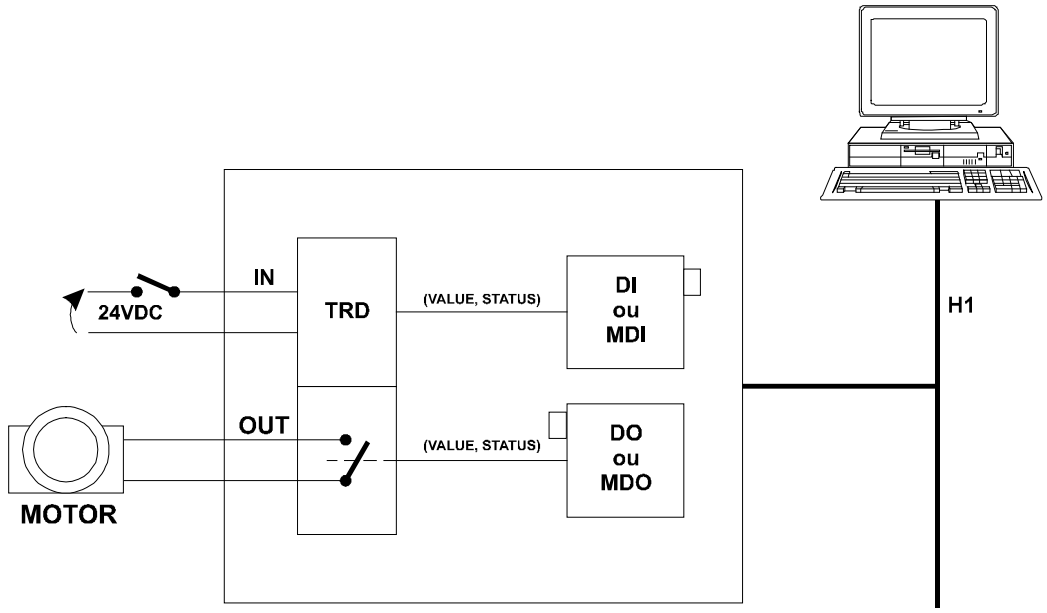


Figure 3.6 - Aplicação 1- DC302

Aplicação 2: Controle Distribuído (O limite de nível baixo acionará um motor, bomba ou uma válvula on/off).

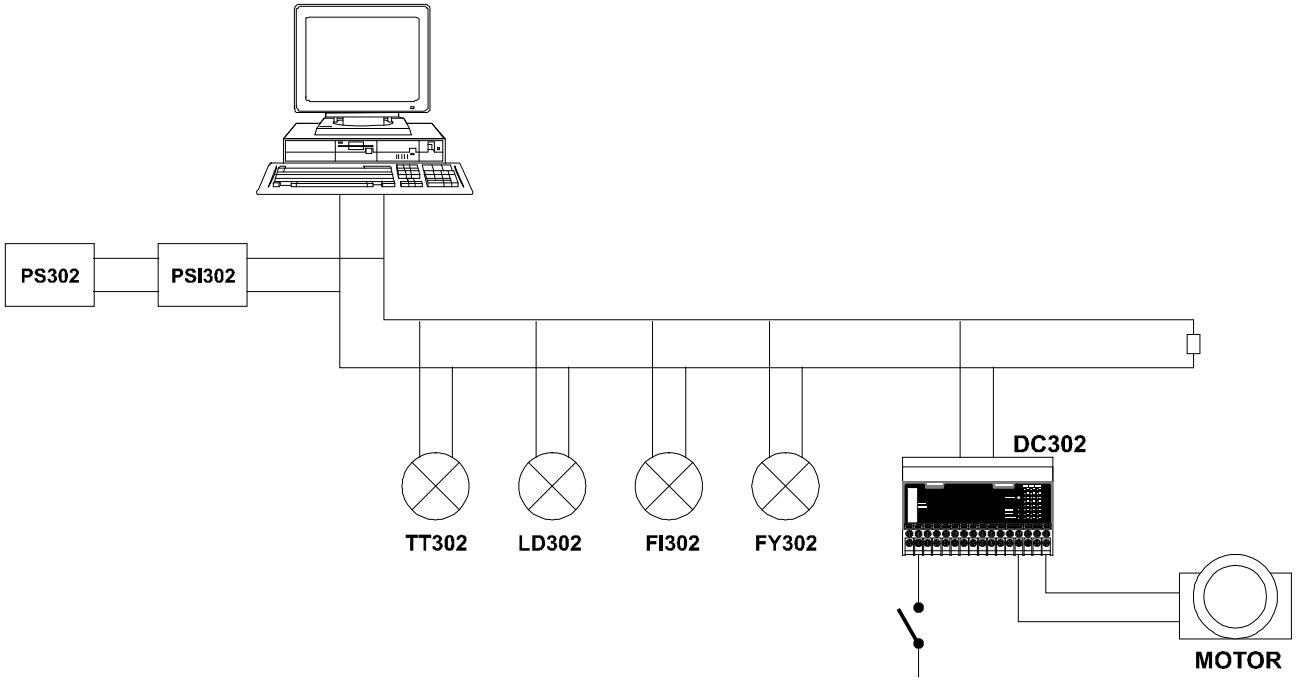


Figure 3.7 - Aplicação 2 - DC302

Aplicação 3: Controle Distribuído (PID step).

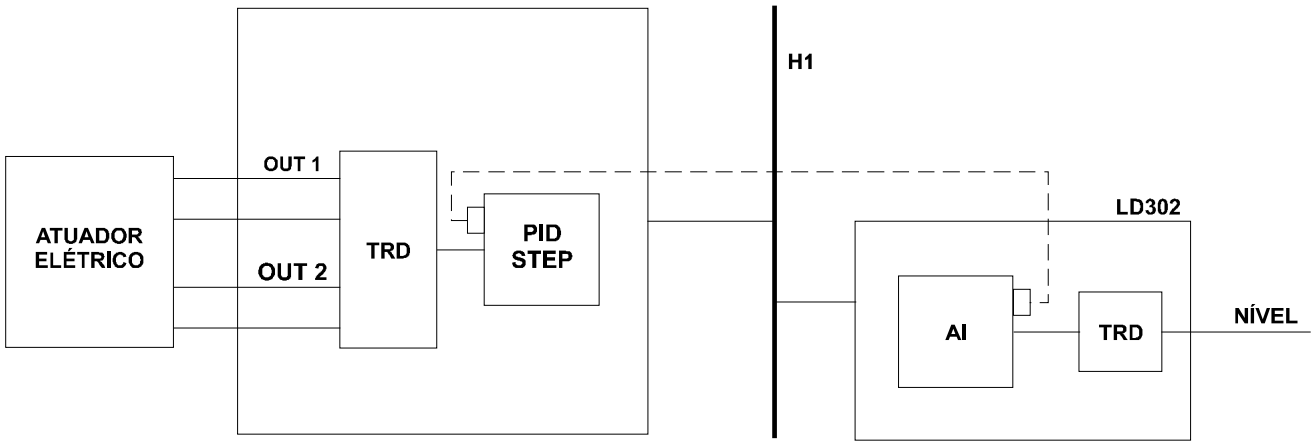


Figure 3.8 - Aplicação 3 - DC302

Aplicação 4: Controle Distribuído discreto usando-se os blocos funcionais TIME e FFET.

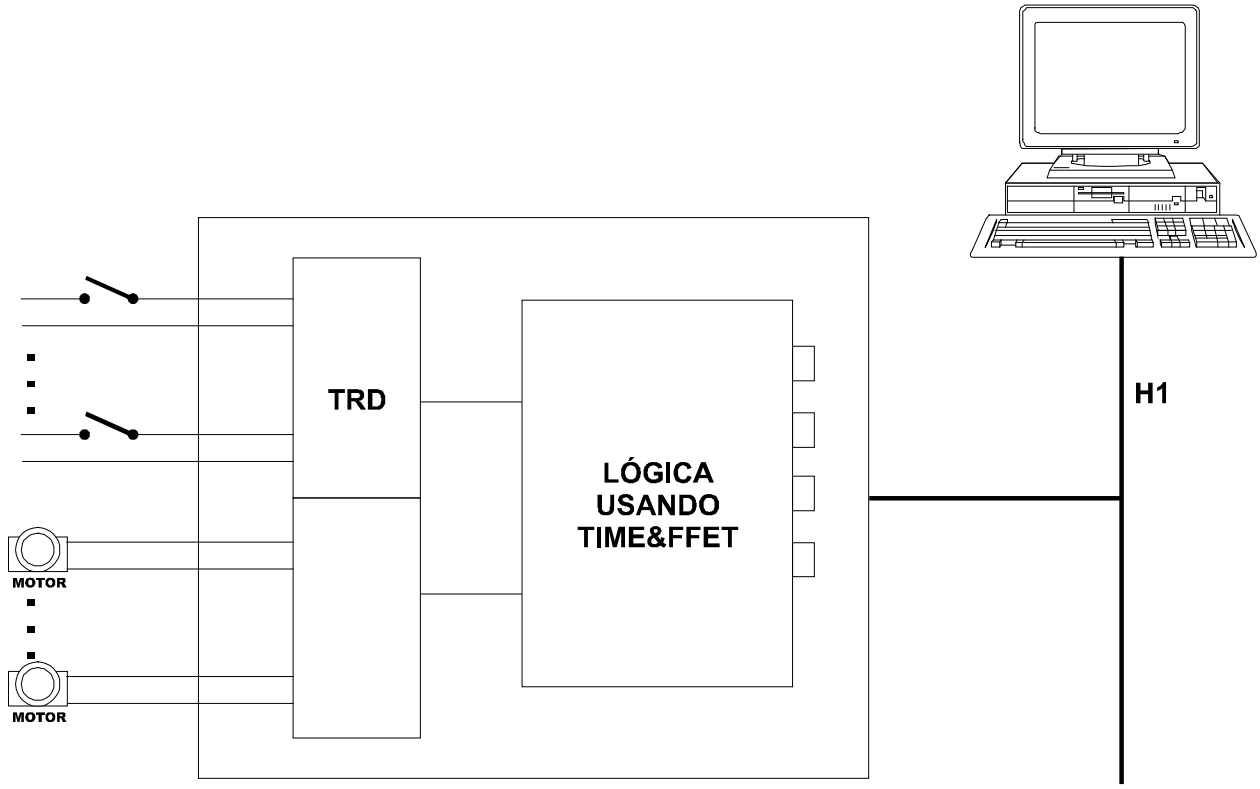


Figure 3.9 - Aplicação 4- DC302

Aplicação 5: Aplicação genérica para o DC302.

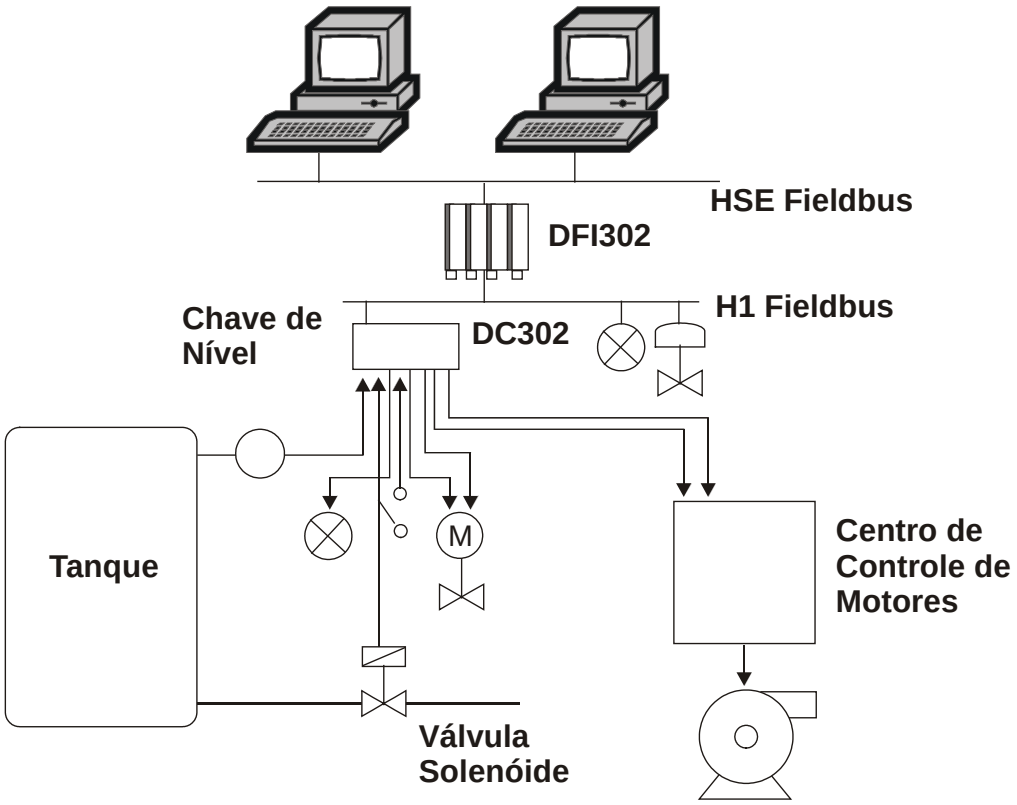


Figure 3.10 - Aplicação 5 - DC302

Bloco Funcional Flexível

Descrição

O bloco FFB pode receber até 8 entradas discretas vindas da rede FF através dos parâmetros IN_D1 a IN_D8 e também disponibilizar 8 saídas discretas à rede FF, através dos parâmetros OUT_D1 a OUT_D8. Pode receber até 16 entradas discretas via hardware (HW_IN) e também disponibilizar 8 saídas discretas de hardware (HW_OUT).

A indicação do estado (status) das entradas depende do sub-sistema de E/S.Os estados das saídas dependem dos cálculos executados pelo bloco.

O Bloco FFB provê lógicas como AND, OR, XOR e NOT e funções como: Timer On-Delay, Timer Off-Delay, Timer Pulse, Pulse Counter Down (CTD), Pulse Counter Up(CTU), Flip-Flop RS e Flip-Flop SR. As lógicas são feitas utilizando-se as entradas discretas (IN_Dx) vindas da rede FF, as saídas disponíveis a rede FF (OUT_Dx), as entradas discretas de hardware (HR_IN), as saídas discretas de hardware (HR_OUT), valores discretos de segurança em condições de falha (FSx) e variáveis discretas auxiliares (AUX's).

BLOCK_ERR

O BLOCK_ERR do bloco FFB refletirá nas seguintes causas:

- Block Configuration Error – o erro de configuração ocorre quando existir um erro na linha de lógica, indicada pelo parâmetro ERROR_LINE e cujo erro é indicado no parâmetro ERROR_CODE.
- Input failure – quando ocorrer falha na fonte de alimentação das entradas.
- Output failure quando ocorrer falha na fonte de alimentação das saídas.
- Out of Service – quando o bloco estiver fora de service (O/S).

Status

O status das saídas OUT_Dx será de acordo com a indicação do parâmetro BLOCK_ERR:

- Outro – ruim: Erro de configuração (Other – Bad: Configuration Error)
- Falha de entrada – ruim: Equipamento em falha (Input failure – Bad: Device Failure)
- Partida – ruim: equipamento em falha (Power up – Bad: Device Failure)

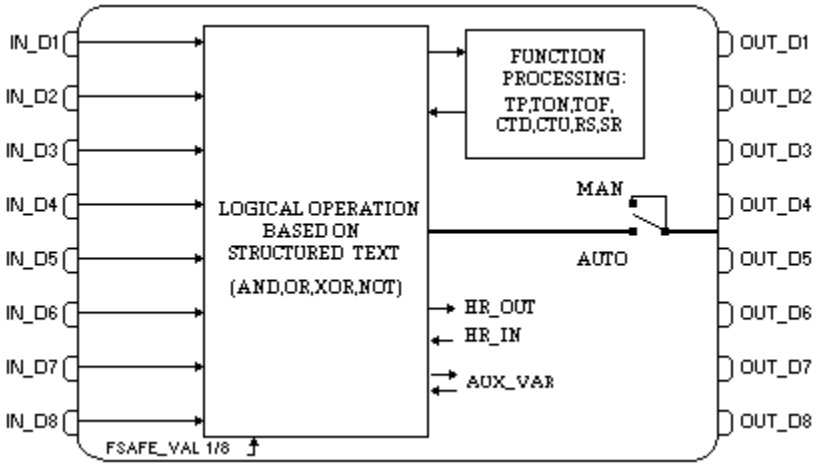
Na execução da lógica, um status maior ou igual a 0x80 é considerado “verdadeiro” (true) e menor que 0x80, é considerado “falso” (false).

Modos Suportados

O/S, MAN e AUTO.

As mudanças nas Linhas Lógicas e seus parâmetros de configuração dependem da seleção de CHANGE_OPTION.

Esquemático



Parâmetros

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Memória / Modo	Descrição
1	ST_REV	Unsigned16		0	Nenhuma	S/RO	O nível de revisão do dado estático associado com o bloco funcional.
2	TAG_DESC	OctString(32)		Spaces	Na	S	Descrição do usuário para a aplicação do bloco.
3	STRATEGY	Unsigned16		0	Nenhuma	S	Pode ser usado para identificar o agrupamento de blocos. Este dado não é verificado ou processado pelo bloco.
4	ALERT_KEY	Unsigned8	1 a 255	0	Nenhuma	S	Número de Identificação da unidade da planta. Esta informação pode ser usada pelo host ou em alarmes.
5	**MODE_BLK**	**DS-69**		**O/S**	**Na**	**S**	**Modo de operação do bloco.**
6	**BLOCK_ERR**	**Bitstring(2)**			**E**	**D / RO**	**Este parâmetro informa condições associadas com o hardware, software e o bloco. Este é um parâmetro bit-string, onde múltiplos erros podem ser mostrados.**
7	PI_POINTER	Unsigned32		0	Nenhuma	S	Índice PI associado ao bloco funcional ou recurso. Um índice zero indica que não existe recurso associado.
8	CONTENTS_REV	Unsigned32		0	Nenhuma	S	Indica a revisão do algorítmo do FFB.
9	**IN_D1**	**DS-66**				**D**	**Entrada discreta nº 1 utilizada pelo cálculo do bloco.**
10	**IN_D2**	**DS-66**				**D**	**Entrada discreta nº 2 utilizada pelo cálculo do bloco.**

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Memória / Modo	Descrição
11	**IN_D3**	**DS-66**				**D**	**Entrada discreta nº 3 utilizada pelo cálculo do bloco.**
12	**IN_D4**	**DS-66**				**D**	**Entrada discreta nº 4 utilizada pelo cálculo do bloco.**
13	**IN_D5**	**DS-66**				**D**	**Entrada discreta nº 5 utilizada pelo cálculo do bloco.**
14	**IN_D6**	**DS-66**				**D**	**Entrada discreta nº 6 utilizada pelo cálculo do bloco.**
15	**IN_D7**	**DS-66**				**D**	**Entrada discreta nº 7 utilizada pelo cálculo do bloco.**
16	**IN_D8**	**DS-66**				**D**	**Entrada discreta nº 8 utilizada pelo cálculo do bloco.**
17	FSTATE_VAL_D1	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 1.
18	FSTATE_VAL_D2	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 2.
19	FSTATE_VAL_D3	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 3.
20	FSTATE_VAL_D4	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 4.
21	FSTATE_VAL_D5	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 5.
22	FSTATE_VAL_D6	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 6.
23	FSTATE_VAL_D7	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 7.
24	FSTATE_VAL_D8	Unsigned8		0		S	Valor discreto utilizado como valor de segurança na condição de falha para a saída de hardware nº 8.
25	**OUT_D1**	**DS-66**				**D / Man**	**Variável discreta da saída 1 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.**
26	**OUT_D2**	**DS-66**				**D / Man**	**Variável discreta da saída 2 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.**
27	**OUT_D3**	**DS-66**				**D / Man**	**Variável discreta da saída 3 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.**
28	OUT_D4	DS-66				D / Man	Variável discreta da saída 4 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.
29	OUT_D5	DS-66				D / Man	Variável discreta da saída 5 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.
30	OUT_D6	DS-66				D / Man	Variável discreta da saída 6 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Memória / Modo	Descrição
31	OUT_D7	DS-66				D / Man	Variável discreta da saída 7 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.
32	OUT_D8	DS-66				D / Man	Variável discreta da saída 8 calculada pelo bloco quando em modo AUTO ou de acordo com o usuário quando em modo MAN.
33	**HW_IN**	**DS-160**				**D / Man**	**Estrutura de dados para as entradas de hardware: 16 unsigned8 para os valores e 1 unsigned8 para o status**
34	**HW_OUT**	**DS-159**				**D / Man**	**Estrutura de dados para as saídas de hardware: 8 unsigned8 para os valores e 1 unsigned8 para o status.**
35	AUX_01_16	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 01_16.
36	AUX_17_32	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 17_32.
37	AUX_33_48	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 33_48.
38	AUX_49_64	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 49_64.
39	AUX_65_80	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 65_80.
40	AUX_81_96	Bitstring(2)				D/ OS	Variável Auxiliar bit enumerated 81_96.
41	TON_PST	16 Floats	Positive	0	Seg	S/ OS	Vetor de 16 elementos em ponto flutuante onde o usuário pode setar em segundos o valor PST para cada Timer ON Delay.
42	TON_CTA	16 Floats		0	seg	D	Vetor de 16 elementos em ponto flutuante onde o usuário pode ver em segundos o valor decorrido para cada Timer ON Delay.
43	TON_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
44	TOFF_PST	16 Floats	Positive	0	Seg	S/ OS	Vetor de 16 elementos em ponto flutuante onde o usuário pode setar em segundos o valor PST para cada Timer OFF Delay.
45	TOFF_CTA	16 Floats		0	seg	D	Vetor de 16 elementos em ponto flutuante onde o usuário pode ver em segundos o valor decorrido para cada Timer OFF Delay.
46	TOFF_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
47	TP_PST	16 Floats	Positive	0	Seg	S/ OS	Vetor de 16 elementos em ponto flutuante onde o usuário pode setar em segundos o valor PST para cada Timer PULSE.
48	TP_CTA	16 Floats		0	Seg	D	Vetor de 16 elementos em ponto flutuante onde o usuário pode ver em segundos o valor decorrido para cada Timer PULSE.
49	TP_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
50	CTU_PST	16 Unsigned32	Positive	0	Nenhuma	S/ OS	Vetor de 16 elementos unsigned integer32 onde pode-se configurar o valor a ser contado para cada contador. O contador irá incrementar de zero ao valor PST.
51	CTU_CTA	16 Unsigned32		0	Nenhuma	D	Vetor de 16 elementos unsigned integer32 onde pode-se acompanhar a contagem parcial para cada contador.
52	CTU_OUT	Bitstring(2)				D	Variável que indica os estados dos contadores ativos.

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Memória / Modo	Descrição
53	CTD_PST	16 Unsigned32	Positive	0	Nenhuma	S/ OS	Vetor de 16 elementos unsigned integer32 onde pode-se configurar o valor a ser contado para cada contador. O contador irá decrementar do valor PST até zero.
54	CTD_CTA	16 Unsigned32		0	Nenhuma	D	Vetor de 16 elementos unsigned integer32 onde pode-se acompanhar a contagem parcial para cada contador.
55	CTD_OUT	Bitstring(2)				D	Variável que indica os estados dos timers ativos.
56	RS_OUT	Bitstring(2)				D	Variável que indica os estados dos Flip-Flops RS ativos.
57	SR_OUT	Bitstring(2)				D	Variável que indica os estados dos Flip-Flops SR ativos.
58	LOGIC_01	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 1.
59	LOGIC_02	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 2.
60	LOGIC_03	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 3.
61	LOGIC_04	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 4.
62	LOGIC_05	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 5.
63	LOGIC_06	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 6.
64	LOGIC_07	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 7.
65	LOGIC_08	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 8.
66	LOGIC_09	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 9.
67	LOGIC_10	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 10.
68	LOGIC_11	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 11.
69	LOGIC_12	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 12.
70	LOGIC_13	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 13.
71	LOGIC_14	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 14.
72	LOGIC_15	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 15.
73	LOGIC_16	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 16.
74	LOGIC_17	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 17.
75	LOGIC_18	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 18.
76	LOGIC_19	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 19.
77	LOGIC_20	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 20.
78	LOGIC_21	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 21.
79	LOGIC_22	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 22.
80	LOGIC_23	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 23.
81	LOGIC_24	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 24.
82	LOGIC_25	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 25.
83	LOGIC_26	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 26.
84	LOGIC_27	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 27.
85	LOGIC_28	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 28.
86	LOGIC_29	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 29.
87	LOGIC_30	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 30.
88	LOGIC_31	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 31.
89	LOGIC_32	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 32.
90	LOGIC_33	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 33.
91	LOGIC_34	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 34.
92	LOGIC_35	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 35.
93	LOGIC_36	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 36.
94	LOGIC_37	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 37.
95	LOGIC_38	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 38.
96	LOGIC_39	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 39.
97	LOGIC_40	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 40.
98	LOGIC_41	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 41.

Idx	Parâmetro	Tipo Dado (comp)	Faixa Válida/ Opções	Valor Default	Unid.	Memória / Modo	Descrição
99	LOGIC_42	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 42.
100	LOGIC_43	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 43.
101	LOGIC_44	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 44.
102	LOGIC_45	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 45.
103	LOGIC_46	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 46.
104	LOGIC_47	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 47.
105	LOGIC_48	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 48.
106	LOGIC_49	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 49.
107	LOGIC_50	VisibleString(24)		Spaces	Na	S/ OS	Linha Lógica de comando 50.
108	**LOGIC_CHECK**	**Unsigned8**	**0 - Enable., 1 – Checked. 2- Changed but not checked yet.**	**1 - Checked.**	**Na**	**D/OS**	**Permite a verificação das linhas lógicas.**
109	**ERROR_LINE**	**Unsigned8**	**0-50**	**1**	**Na**	**S**	**Indica a linha onde se tem erro. Valor igual a zero indica que não existe linha com erro.**
110	ERROR_CODE	Unsigned8	0 - Logic Ok. 1 - Exceed String Length or string not valid. 2 - Non valid operand. 3 - No implemented logic or missing ';' 4 - Missing parentheses or argument not valid. 5 - Non valid resource. 6 - Argument not valid. 7 - Function not valid 8 - Non available resource. 9 - Non valid attribution. 10 - First Argument not valid. 11- Second Argument not valid.	3 - No implemented logic or missing ';'	Na	S	Indica o código de erro.
111	CHANGE_OPTION	Unsigned8	0 - Logic parameter changes are only allowed in Out of Service. 1 - Always accept Logic parameter changes.	0 - Logic parameter changes are only allowed in Out of Service.	Na	S	Habilita mudanças nas linhas de lógicas independentes do modo de operação do bloco.
112	UPDATE_EVT	DS-73			Na	D	Este alerta é gerado por qualquer mudança no dado estático.
113	BLOCK_ALM	DS-72			Na	D	Ë usado para indicar falhas de hardware, configuração ou problemas com o bloco.

Legenda:

E – Lista de Parâmetros;
Na – Parâmetro Admensional;
RO – Somente Leitura;
D – dinâmico;
N – não volátil;
S – estático;
Linha com Preenchimento de Fundo Cinza: Parâmetros Default do Syscon

A tabela, a seguir, descreve as Operações Lógicas e os Comandos de Linha e seus Símbolos Correspondentes usados nas linhas lógicas:

Operação Lógica e Comando de Linha	Símbolo - descrição
E	&
OU	|
OU Exclusivo	^
Inversor	!
Igual	=
(arg1,arg2)	Argumentos de função
;	Finalizador de linha

A lógica Inversora (!) trabalha somente com variáveis simples. Exemplo: OUT1=!IN1;
Note que não é permitido ter, por exemplo, OUT1=!TP01(IN1);. Para se trabalhar desta forma, deve-se fazer: A01= TP01(IN1);. -> OUT1=!A01;

A execução da linha lógica é sempre executada linha por linha e da esquerda para a direita. Não são permitidos espaços entre caracteres. **Não é permitido linhas vazias entre linhas com lógicas implementadas, isto é, a implementação deve ser em seqüência.**
Depois de escrever a lógica dentro dos parâmetros LOGIC_XX (XX:01 -> XX:50), o usuário precisa selecionar a opção “Enable” no parâmetro LOGIC_CHECK para verificar se há erros de sintaxe. **Quando se utiliza o processo de download, é primordial configurar, primeiramente, os parâmetros LOGIC_XX (XX:01 -> XX:50) e depois o parâmetro LOGIC_CKECK. Esta seqüência é fundamental para executar a verificação.** A tabela abaixo mostra o mnemônico para cada parâmetro de bloco usado nas linhas lógicas. O mnemônico deve estar em letras maiúsculas:

Parâmetro	Mnemônico
HW_IN.Value1	I01
HW_IN.Value2	I02
HW_IN.Value3	I03
HW_IN.Value4	I04
HW_IN.Value5	I05
HW_IN.Value6	I06
HW_IN.Value7	I07
HW_IN.Value8	I08
HW_IN.Value9	I09
HW_IN.Value10	I10
HW_IN.Value11	I11
HW_IN.Value12	I12
HW_IN.Value13	I13
HW_IN.Value14	I14
HW_IN.Value15	I15
HW_IN.Value16	I16
HW_IN.Status	SI
HW_OUT.Status	SO
HW_OUT.Value1	O1
HW_OUT.Value2	O2
HW_OUT.Value3	O3
HW_OUT.Value4	O4
HW_OUT.Value5	O5
HW_OUT.Value6	O6
HW_OUT.Value7	O7
HW_OUT.Value8	O8
IN_D1.Status	IN1S
IN_D2.Status	IN2S
IN_D3.Status	IN3S
IN_D4.Status	IN4S
IN_D5.Status	IN5S
IN_D6.Status	IN6S
IN_D7.Status	IN7S
IN_D8.Status	IN8S
IN_D1.Value	IN1
IN_D2.Value	IN2
IN_D3.Value	IN3
IN_D4.Value	IN4
IN_D5.Value	IN5
IN_D6.Value	IN6
IN_D7.Value	IN7
IN_D8.Value	IN8
OUT_D1.Status	SOUT1
OUT_D2.Status	SOUT2
OUT_D3.Status	SOUT3
OUT_D4.Status	SOUT4
OUT_D5.Status	SOUT5

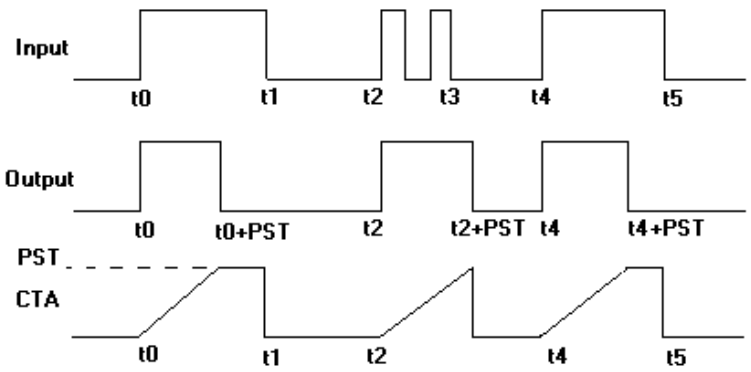
Parâmetro	Mnemônico
OUT_D6.Status	SOUT6
OUT_D7.Status	SOUT7
OUT_D8.Status	SOUT8
OUT_D1.Value	OUT1
OUT_D2.Value	OUT2
OUT_D3.Value	OUT3
OUT_D4.Value	OUT4
OUT_D5.Value	OUT5
OUT_D6.Value	OUT6
OUT_D7.Value	OUT7
OUT_D8.Value	OUT8
FSTATE_VAL_D1	FS1
FSTATE_VAL_D2	FS2
FSTATE_VAL_D3	FS3
FSTATE_VAL_D4	FS4
FSTATE_VAL_D5	FS5
FSTATE_VAL_D6	FS6
FSTATE_VAL_D7	FS7
FSTATE_VAL_D8	FS8
AUX_01_16	A01-A16
AUX_17_32	A17-A32
AUX_33_48	A33-A48
AUX_49_64	A49-A64
AUX_65_80	A65-A80
AUX_81_96	A81-A96
TON	TON01-TON16
TOFF	TOF01-TOF16
TP	TP01-TP16
CTU	CTU01-CTU16
CTD	CTD01-CTD16
RS	RS01-RS16
SR	SR01-SR16

Funções

Para cada tipo de função há 16 recursos disponíveis e o usuário pode usar somente um recurso de cada vez. O usuário poderá atribuir o resultado de uma função aos bits discretos auxiliares.

TP TIMER PULSE

Esta função gera em sua saída um pulso de largura de tempo fixa a cada transição de falso para verdadeiro (borda de subida) na entrada. A largura de pulso pode ser programada no parâmetro TP_PST, em segundos. Transições na entrada serão ignoradas, enquanto a saída estiver ativa. O timer corrente pode ser monitorado no parâmetro TP_CTA.



Função Timer Pulse – diagramas de tempo

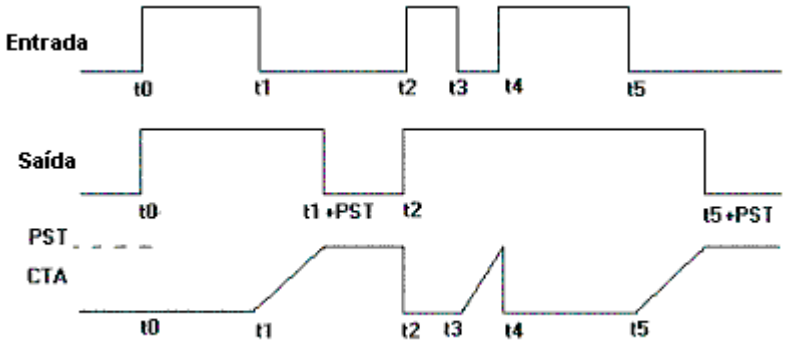
A sintaxe para a função Timer Pulse é: TPxx(arg)
Onde, xx é o recurso de 01 a 16 e arg é o argumento da função e deve ser uma variável simples. Exemplos:

O1=TP01(IN1);
OUT1= TP01(A05);
OUT3=TP08(FS1);

Por exemplo, os exemplos seguintes são permitidos na linha lógica:
O1=TP01(IN1&IN2);: note que o argumento é o resultado de uma operação e isto não é permitido.
O1=TP10(!IN1);: note que o argumento é o resultado de função NOT e isto não é permitido.
O1=TP10(CTD01(IN1,IN2));: note que o argumento é o resultado de uma função e isto não é permitido.

TON TIMER ON-DELAY

Nesta função a saída vai para verdadeiro após um período de tempo em segundos configurável no parâmetro TP_PST, após a entrada ir para verdadeiro. Se a entrada for para falso antes do tempo PST, a saída permanecerá em falso. O parâmetro CTA mostra o tempo decorrente até o valor PST.



Função Timer On-Delay– diagramas de tempo

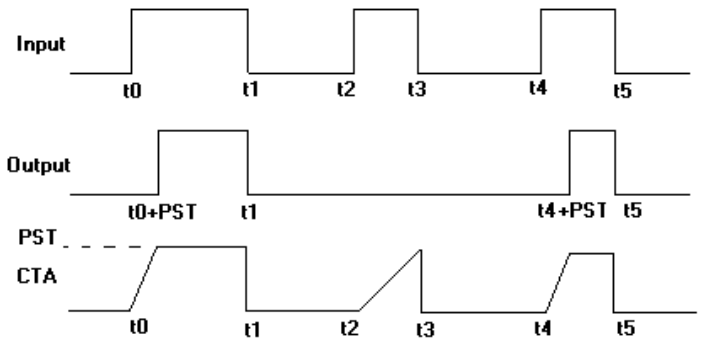
A sintaxe para a função Timer On-Delay é: TONxx(arg)
Onde, xx é o recurso de 01 a 16 e arg é o argumento da função e deve ser uma variável simples. Exemplos:

O1=TON01(IN1)&SI;
OUT1= TON01(A05);
OUT3=TON08(FS1);
Por exemplo, os seguintes exemplos não são permitidos na linha lógica:
O1=TON01(IN1&IN2);: note que o argumento é o resultado de uma operação e isto não é permitido.

O1=TON10(!IN1);: note que o argumento é o resultado de uma função NOT e isto não é permitido.
O1=TON10(CTD01(IN1,IN2));: note que o argumento é o resultado de uma função e isto não é permitido.

TOF TIMER OFF-DELAY

Esta função estende o estado verdadeiro da entrada por um período de tempo em segundos configurado via parâmetro TOF_PST. Se a entrada vai para verdadeiro antes da saída ir para falso, a saída ficará em verdadeiro até que a saída vá para falso e decorra o tempo PST. O parâmetro CTA mostra o tempo corrente.



Função Timer OFF-Delay – Diagramas de Tempo

A sintaxe para Timer Off-Delay é: TOFxx(arg)
Onde, xx é o recurso usado de 01 a 16 e arg é o argumento da função e deve ser uma variável simples. Exemplos:

O1=TOF01(IN1)&SI;
OUT1= TOF01(A05);
OUT3=TOF08(FS1);

Por exemplo, os seguintes exemplos são não permitidos na linha lógica:
O1=TOF01(IN1&IN2);: note que o argumento é o resultado de uma operação e isto não é permitido.
O1=TOF10(!IN1);: note que o argumento é o resultado de uma função e isto não é permitido.
O1=TOF10(CTD01(IN1,IN2));: note que o argumento é o resultado de uma função e isto não é permitido.

CTD PULSE COUNTER DOWN

Esta função é usada para contar transições de subida (de falso para verdadeiro) da entrada (arg1). A cada transição de subida, o parâmetro CTA decrementa de uma unidade. Quando o CTA atinge o valor zero, a saída do contador irá para verdadeiro. O valor do contador CTA será carregado com o valor de PST. Uma transição de falso para verdadeiro no argumento 2 (arg2) coloca a saída em falso e o parâmetro CTA receberá o valor PST.

A sintaxe para CTD é: CTDxx(arg1,arg2)
Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são argumentos das funções e devem ser variáveis simples. Exemplos:
O3=CTD10(IN1,IN2);
OUT1=CTD03(A11,A14)&SI;

Por exemplo, os seguintes exemplos são não permitidos na linha de lógica:
O1=CTD01(IN1&IN2,IN3);: note que o argumento é o resultado de uma operação e isto não é permitido.
O1=CTD10(!IN1,IN3);: note que o argumento é o resultado de uma função e isto não é permitido.
O1=CTD10(TP01(IN1),IN2);: note que o argumento é o resultado de uma função e isto não é permitido.

CTU PULSE COUNTER UP

Esta função é usada para contar transições de subida (de falso para verdadeiro) na entrada (arg1). A cada transição o parâmetro CTA incrementará de uma unidade. Ao atingir o valor PST, a saída desta função irá para o estado lógico verdadeiro e o valor CTA será carregado com valor zero. Uma transição de falso para verdadeiro no argumento 2 (arg2) coloca a saída em falso e o parâmetro CTA receberá o valor zero.

A sintaxe para CTU é: CTUxx(arg1,arg2)

Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são os argumentos da função e devem ser variáveis simples. Exemplos:

O3=CTU10(IN1,IN2);
OUT1=CTU03(A11,A14)&SI;

Por exemplo, os seguintes exemplos são não permitidos na linha de lógica:
O1=CTU01(IN1&IN2,IN3);: note que o argumento é o resultado de uma operação e isto não é permitido.
O1=CTU10(!IN1,IN3);: note que o argumento é o resultado de uma função e isto não é permitido.
O1=CTU10(TP01(IN1),IN2);: note que o argumento é o resultado de uma função e isto não é permitido.

FLIP-FLOP RS

Esta função tem a seguinte tabela de operação:

R(arg1)	S(arg2)	OUT
0	0	Último estado
0	1	1
1	0	0
1	1	0

A sintaxe para RS Flip-Flop é: RSxx(arg1,arg2)

Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são os argumentos da função e devem ser variáveis simples. Exemplos:

O3=RS10(IN1,IN2);
OUT1=RS03(A11,A14)&SI;

Por exemplo, os seguintes exemplos são não permitidos na linha de lógica:
O1=RS01(IN1&IN2,IN3);: note que o argumento é o resultado de uma operação e isto não é permitido.
O1=RS10(!IN1,IN3);: note que o argumento é o resultado de uma função e isto não é permitido.
O1=RS10(TP01(IN1),IN2);: observe que o argumento é o resultado de uma função e isto não é permitido.

FLIP-FLOP SR

Esta função tem a seguinte tabela de operação:

S(arg1)	R(arg2)	OUT
0	0	Último Estado
0	1	0
1	0	1
1	1	1

A sintaxe para SR Flip-Flop é: SRxx(arg1,arg2)

Onde, xx é o recurso de 01 a 16 e arg1 e arg2 são os argumentos da função e devem ser variáveis simples. Exemplos:

O3=SR10(IN1,IN2);
OUT1=SR03(A11,A14)&SI;

Por exemplo, os seguintes exemplos são não permitidos na linha de lógica:
O1=SR01(IN1&IN2,IN3);: o argumento é o resultado de uma operação e isto não é permitido.
O1=SR10(!IN1,IN3);: o argumento é o resultado de uma função NOT e isto não é permitido.
O1=SR10(TP01(IN1),IN2);: o argumento é o resultado de uma função e isto não é permitido.

Códigos de Erros

Alguns exemplos de condições de erro:

Error Code: "Exceed String Length or string not valid."
a) OUT1=IN1&IN2&IN2|IN4^IN5|IN6;

Note que exitem 29 caracteres no texto e o máximo permitido é 24.

b) OUT1=IN1&in2;

Note que todos os mnemônicos devem estar em Letras Maiúsculas.

Error Code: "Non valid operand."
OUT1=IN1%IN2;

Note que o símbolo % não é permitido. Veja também a tabela que descreve a os Operadores Lógicos e as Linhas de Comando.

Error Code: "No implemented logic or missing ';'."
OUT1=IN1

Note que falta o terminador ";" no final da linha de lógica.

Error Code: "Missing parentheses or argument not valid."
OUT1=TP10(IN1;

Observe que falta um parênteses na função timer pulse.

Error Code: "Non valid resource."
OUT1=TP18(IN1);

Observe que há até 16 recursos para cada função.

Error Code: "Argument not valid."
OUT1=TP10(IN10);

Observe que há somente 8 entradas discretas vindas da rede FF. IN10 não é um argumento válido.

Error Code: "Function not valid."
OUT1=TR10(IN1);

Observe que TR não é uma função válida.

Error Code: "Non available resource."
OUT1=TP10(IN1);

A03=TP10(IN7);

Observe que há 16 recursos para cada função e o recurso 10 já foi utilizado e não pode ser usado novamente em uma lógica posterior. O que pode ser feito é atribuir o resultado da função a uma variável auxiliar e esta ser usada várias vezes.
A03=TP10(IN7);

Error Code: "Non valid attribution."
IN1=IN2^TP03(IN4);

Observe que não é permitida atribuição para entradas.

Error Code: "First Argument not valid."
OUT1=CTD01(!IN1,IN2);

Observe que os argumentos devem ser necessariamente variáveis simples e não resultados de lógica ou funções.

OUT1=RS11(IN15,IN2);

Observe que o primeiro argumento da função não é um recurso válido.

Error Code: "Second Argument not valid."
a) OUT1=CTD01(IN1,!IN2);

Note que necessariamente os argumentos de função devem ser variáveis simples e não resultados de lógicas ou funções.

OUT1=RS11(IN1,IN20);

Observe que o segundo argumento da função não é um recurso válido.

Exemplos de aplicações

1) De acordo com a próxima figura, temos uma aplicação industrial onde se tem o enchimento de garrafas com fluido químico. A esteira é movimentada até que um sensor detecte sua presença. A esteira deve parar e abrir a vávula de enchimento até que seja detectado o nível pelo sensor de nível.Depois de detectar o nível, o sistema deve esperar durante 10 segundos e mover a esteira novamente até a próxima garrafa.

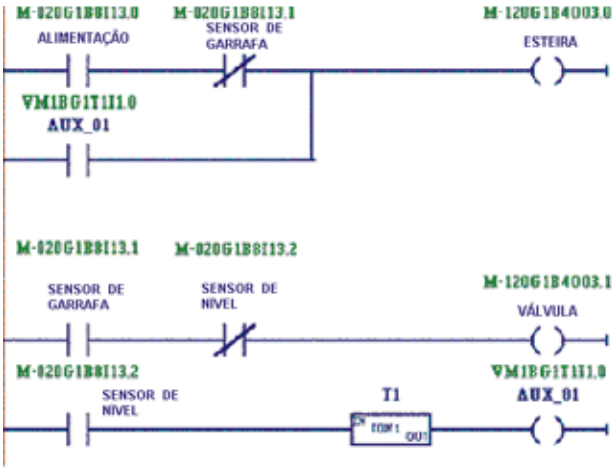
Usando o bloco funcional Flexível, temos as seguintes definições:

- ☑ A esteira será ligada utilizando a saída 01 de hardware (O1);
- ☑ A válvula de enchimento será ligada utilizando a saída 02 de hardware (O2);
- ☑ sensor de garrafa será conectado a entrada de 01 de hardware (I01);
- ☑ sensor de nível será conectado a entrada de 02 de hardware (I02);
- ☑ A alimentação do sistema será a entrada de 03 de hardware (I03);

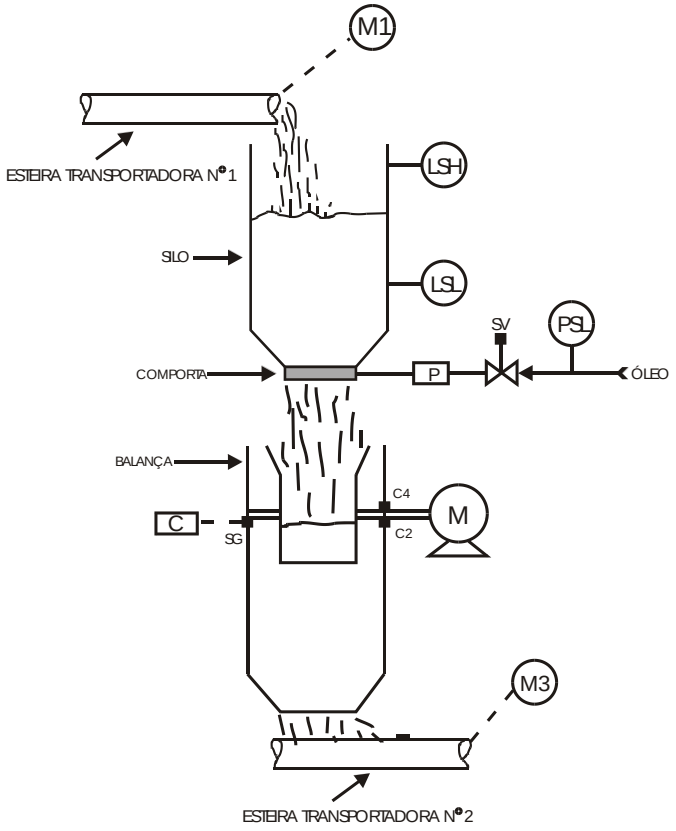
Tem-se, então, a seguinte configuração:

TON_PST resource [01] = 10.0s.
LOGIC_01 A01=TON01(I02);
LOGIC_02 O1=I03&!I01|A01;
LOGIC_03 O2=I01&!I02;

Fazendo analogia com a programação ladder, tem-se:



2) Na seguinte aplicação mecânica, tem-se o controle de passos para operar uma balança eletromecânica, que efetua a pesagem de rocha fosfatada.
O peso do processo é feito por batelada, o sistema executa um ciclo completo de pesagem a cada intervalo de 20 segundos. Veja a figura a seguir:



M1 e M3 - Motores acionadores das esteiras
C2 e C4 – Chaves fim de curso
LSH - Sensor de Nível Alto
LSL - Sensor de Nível Baixo
SG - Célula de Carga
SV - Válvula Solenóide
M - Motor acionador de Caçamba
P - Pistão acionador de Comporta
C - Circuito de Pesagem

Processo:

O sistema necessita das seguintes condições para fazer o startup:

- ☑ Nível da pedra fosfática (LSL desativado);
- ☑ Pressão do Óleo (PSL ligado);
- ☑ Esteira transportadora nº 2 em movimento (M3 ligado);
- ☑ Caçamba na posição inicial (C4 ligado);

Satisfeitas as condições iniciais, observa-se que:

- ☑ Acionando-se o botão de partida, a comporta abre-se, dando início ao carregamento da caçamba.
- ☑ Uma vez atingido o peso desejado, a comporta fecha. Decorridos 5 segundos a caçamba efetua um giro de 180º , descarregando o produto na esteira transportadora número 2.

Observação:

- Esta nova posição será detectada por C2 e após 5 segundos, a caçamba retornará à posição inicial e isto será detectado pelo C4.
- Depois da caçamba retornar para a posição inicial, ter-se-á um novo ciclo de pesagem.

Comentário:

- A seqüência de operação deve ser interrompida se qualquer um dos pré-requisitos não for satisfeito.
- A comporta silo é acionada por um pistão hidráulico.

Usando o Bloco Funcional Fexível, tem-se as seguintes definições:

- ☑ LSL será conectado à entrada do hardware 01 (I01);
- ☑ LSH será conectado à entrada do hardware 02 (I02);
- ☑ PSL será conectado à entrada do hardware 03 (I03);
- ☑ C2 será conectado à entrada do hardware 04 (I04);
- ☑ C4 será conectado à entrada do hardware 05 (I05);
- ☑ Alimentação será conectada à entrada do hardware 06 (I06);
- ☑ M3 será conectado à entrada do hardware 07 (I07);
- ☑ M será conectado à saída do hardware 01 (O1);
- ☑ A Comport será ativada pela saída do hardware 02 (O2);
- ☑ M1 será ativado pela saída do hardware 03 (O3);

Tem-se a seguinte configuração:

TON_PST recurso [01] = 5.0s.
LOGIC_01 A01=!I01&I03&I07&I05;
LOGIC_02 A02=I06&RS01(I02,I01);
LOGIC_03 O3=A02&I03;
LOGIC_04 A03=I03&I07;
LOGIC_05 O2=I06&A03&!I04;
LOGIC_06 O1=TON01(I04)&!I05&A03;

3) Usando Valores de Estado de Falha:

Supondo que exista a seguinte condição:

- ☑ A01: recebe a lógica entre os status das entradas discretas 1 e 2, como esta: A01=IN1S&IN2S; quando o status é ruim, uma destas entradas, então A01=false(0), de outra forma, A01=true (1);
- ☑ FS1: é o valor de segurança de falha para O1;
- ☑ A02: é o bit que contém a lógica para O1;

Há a seguinte tabela entre FS1, A01 e A02:

FS1	A01	A02	O1
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Então,
A03=!FS1&A01&A02;
A04=FS1&!A01&!A02;
A05=FS1&!A01&A02;
A06=FS1&A01&A02;
O1=A03|A04|A05|A06;

Seção 4

PROCEDIMENTOS DE MANUTENÇÃO

Geral

O **DC302 E/S Remotas Fieldbus** foi extremamente testado e inspecionado antes de ser entregue ao usuário. Entretanto, durante o seu desenvolvimento, foi dada a possibilidade de reparos pelo usuário, quando necessário.
Em geral, é recomendado que não se repare as placas eletrônicas. Ao invés disso, o usuário deve ter partes sobressalentes, as quais podem ser adquiridas com a SMAR quando necessárias.

PROBLEMAS E SOLUÇÕES	
Sintoma	**Provável fonte do problema**
Sem Corrente Quiescente	*Conexão de E/S do DC302:* *Verifique conexões segundo a polaridade e continuidade.* *Fonte de Alimentação:* *Verifique o sinal de saída da fonte de alimentação. A tensão de alimentação na borneira do DC302 deve estar entre 9 e 32 Vdc.* *Falha no circuito eletrônico:* *Verifique as placas eletrônicas, trocando-as pelas sobressalentes.*
Sem Comunicação	Conexões com a rede de trabalho: Verifique as conexões com a rede, como os equipamentos, fontes de alimentação e terminadores. Impedância da rede: Verifique a impedância da rede (da fonte de alimentação e dos terminadores). Configuração do Mestre: Verifique a configuração de comunicação e parametrização do mestre. Configuração da rede: Verifique a configuração de comunicação na rede de trabalho. *Falha no circuito eletrônico:* *Verifique as placas eletrônicas, trocando-as pelas sobressalentes.*
Entradas Incorretas	Conexão dos terminais de entrada: Verifique a polaridade e continuidade. Fonte de alimentação das entradas: *Verifique a alimentação. A tensão deve estar entre 18 e 30 VDC e o consumo típico quando todas as entradas estão ativas é 120mA.*
Saídas Incorretas	Conexão dos terminais de saída: Verifique polaridade e continuidade. Fonte de alimentação das saídas: *Verifique a alimentação. A tensão deve estar entre 20 e 30 VDC e a máxima corrente de saída é de 0.5 A.*

Procedimento de desmontagem

Refira-se a Figura 4.1 - Vista Explodida do DC302. Certifique-se de que tenha desconectado a fonte de alimentação antes de desmontar o DC302.

ATENÇÃO
As placas possuem componentes CMOS que podem ser danificados por descargas eletrostáticas. Observe os procedimentos corretos para manipular estes componentes. Também é recomendado armazenar as placas de circuito em embalagens à prova de descargas eletrostáticas.

Solte as travas laterais que prendem a caixa principal do invólucro e, então, a trava principal. A placa principal e a de E/S poderão, então, ser acessadas.
Para removê-las, retire os parafusos que fixam-nas e manuseie cuidadosamente, sem danificá-las.

Procedimento de Montagem

- Coloque as placas cuidadosamente em suas posições no invólucro.
- Aperte os parafusos de fixação das mesmas.
- Certifique-se de que as conexões entre as mesmas estejam corretas.
- Observe a posição dos LEDs e cuidadosamente encaixe a tampa principal, travando-a lateralmente e depois através da trava principal.

Procedimentos de Atualização do Firmware do DC302

Para atualizar o firmware do DC302 refira-se ao manual da FDI302, disponível no website da Smar: www.smar.com.br .

Acessórios

A placa principal e a de E/S podem ser trocadas independentemente.

ACESSÓRIOS	
Código de Pedido	**Descrição**
BC302	Interface Fieldbus/RS232
SYSCON	Ferramenta de Configuração
PS302	Fonte de Alimentação
PSI302	Impedância da Fonte de Alimentação
BT302	Terminador
PCI	Interface de Controle de Processo
FDI-302-2	Interface de Dispositivo de Campo Foundation Fieldbus & Profibus PA

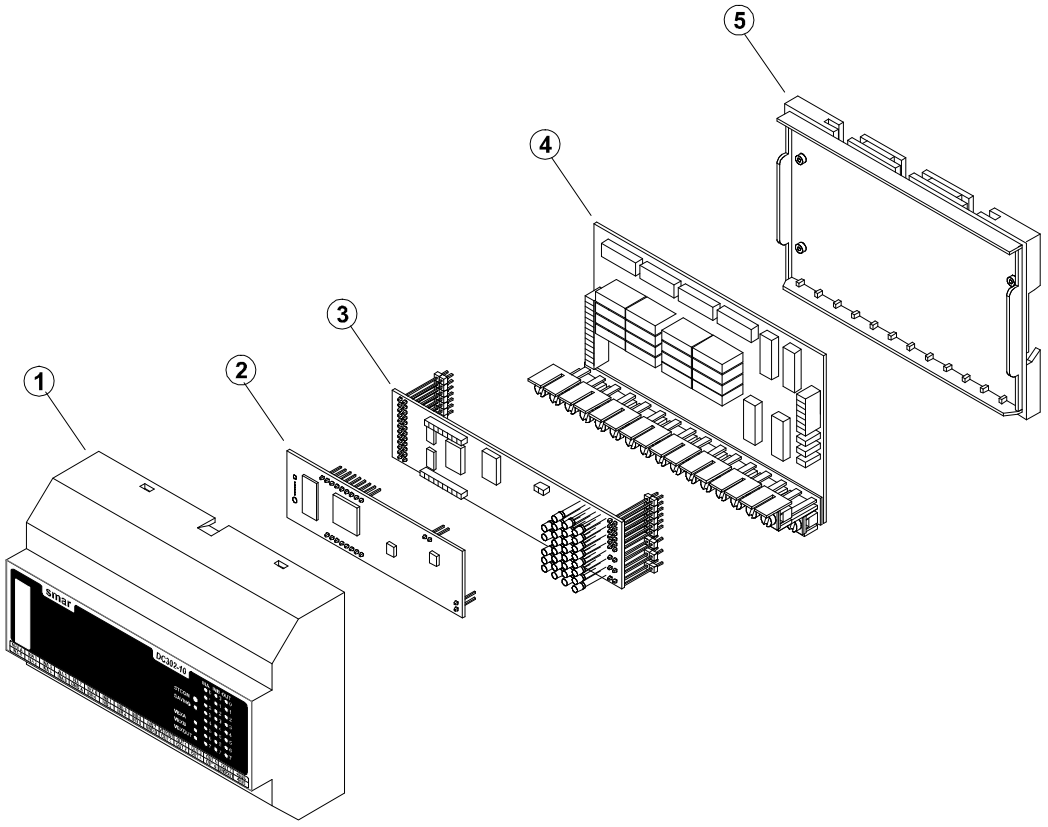


Figure 4.1 – Vista Explodida do DC302

Partes Sobressalentes

NOME	POSIÇÃO	CÓDIGO
Invólucro	*1 e 5*	400 - 0367
Placa Eletrônica Principal	*2*	400 - 0368
Placa de Interface	*3*	400 - 0369
Placa de E/S	*4*	400 - 0370

Seção 5

CARACTERÍSTICAS TÉCNICAS

Geral

Sinal (Comunicação)	Digital. Modo tensão Fieldbus 31,25 Kbits/s.
Fontes de Alimentação	Onde se requer isolação entre as fontes das entradas e saídas, recomenda-se usar no mínimo duas fontes de alimentação, sendo uma para as entradas e outra para as saídas e VDC. Onde a aplicação não exigir isolação entre as fontes, pode-se ter somente uma fonte para entradas, saídas e VDC. As saídas e entradas são isoladas opticamente.
Consumo de corrente quiescente	150 mA da fonte VDC
Corrente de Partida	400 mA durante os primeiros 20s após a partida.
Tempo de partida	Aproximadamente 10 segundos.
Tempo de atualização	Aproximadamente 100 ms. O tempo de atualização está relacionado com a atualização das entradas e saídas do DC302, não depende do macrocycle do sistema.
Impedância de saída	Não Intrinsecamente seguro de 7,8 kHz - 39 kHz deve ser maior ou igual a 3 kΩ. Intrinsecamente seguro: (assumindo barreira de SI na alimentação) de 7,8 kHz – 39 kHz maior ou igual a 3 kΩ.
Efeito de Vibração	De acordo com SAMA PMC 31.1.
Limites de Temperatura	**Operação:** -40 a 85ºC (-40 a 185ºF). **Armazenagem:** -40 a 110ºC (-40 a 230ºF).
Invólucro	**Invólucro/base**: Policarbonato, 10% fibra de vidro. **Terminais:** Parafusos em Zinco, aço cromado. **Faixa de Temperatura** 110ºC (230 ºF) UL94VO. **Proteção:** IP20 (toque com o dedo) e VBG4 e outros requisitos europeus de prevenção de acidentes. Pode ser opcionalmente fornecido em caixa de distribuição a prova de explosão para montagem no campo.
Montagem	Usando trilho DIN (TS35-DIN EN 50022 ou TS32-DIN EN50035 ou TS15-DIN EN50045).

Entradas do DC302

Descrição das Entradas

As entradas recebem tensões DC e convertem em sinal lógico Ligado ou Desligado.
Possui um grupo de 16 entradas isoladas oticamente que recebem 24Vdc.

Na falha da fonte de alimentação das entradas haverá indicação no BLOCK_ERR dos blocos funcionais de entrada, tais como DI, MDI, FFB (ver manual de Function Blocks).

Especificações Técnicas

Arquitetura	O número de entradas é 16.
Isolação, os grupos são individualmente isolados	Isolação Ótica de 5000 Vac.
Fonte Externa	18 - 30 Vdc.
Consumo Típico do grupo	120 mA (todas as entradas em estado ligado).
Indicador de Alimentação	LED verde.
Entradas	Nível no estado LIGADO (Verdadeiro Lógico) 15 - 30 Vdc.
	Nível no estado DESLIGADO (Falso Lógico) 0 - 5 Vdc.
Impedância Típica	3,9 kΩ.
Indicador do Estado de ativação	LED vermelho.
Informação de Chaveamento	Tempo de “0” a “1”: 30 μs.
	Tempo de “0” a “1”: 50 μs.
Cabeamento	Fio único: 14 AWG (2 mm 2) 14 AWG (2 mm2).
	Dois fios: 20 AWG (0,5 mm2).

DC302: Saídas em Coletor Aberto

Descrição - Saídas

As saídas estão projetadas com transistores na configuração NPN, coletor aberto de forma a trabalhar com relés, solenóides e outras cargas DC com 0,5 A por saída. Todo o grupo de saídas compartilha o mesmo terra e elas são isoladas uma das outras e da rede Fieldbus.

Na falha da fonte de alimentação das saidas haverá indicação no BLOCK_ERR dos blocos funcionais de saida, tais como DO, MDO, FFB, STEP_PID (ver manual de Function Blocks).

NOTA
As saídas discretas podem ser atuadas por vários blocos como o DO, MDO, STEP_PID, FFB. Porém, deve-se atentar que a saída ou o conjunto que for utilizado para um bloco não deve ser utilizado para outro bloco.

Especificações Técnicas

Arquitetura	Número de saídas: 8.
Isolação	Isolação Ótica de 5000 Vac.
Fonte Externa	20 a 30 Vdc.
Consumo Máximo	35 mA.
Indicação de Alimentação	LED verde.
Saídas	Tensão máxima chaveada: 30 Vdc.
	Tensão Máxima de Saturação 0,55 V a 0,5 A.
	Máxima Corrente por saída : 0,5 A.
	Indicação de saída ativa: LED vermelho.
	Lógica da indicação: Ligado quando o transistor estiver ligado.
	Máxima Corrente de "Leakage": 100 μA a 35 Vdc.
Condição das Saídas durante: **Partida (Power-Up)** **Atualização do Firmware** **Download da Configuração**	DESLIGADO.
Proteção independente das saídas	Desligamento Térmico: 165^{o}C.
	Histerese Térmica 15^{o}C.
	Proteção de sobre-corrente: 1,3 A a 25 Vdc máximo.
Diodo Clamp, informação de chaveamento	Tempo de 0 a 1: 250 μs.
	Tempo de 1 a 0: 3 μs.
Cabeamento	Fio único: 14 AWG (2 mm^2).
	Dois fios: 20 AWG (0,5 mm^2).

Código de Pedido

MODELO	DESCRIÇÃO
DC302-10 E/S Remotas Fieldbus	1 grupo de 16 entradas a 24Vdc isoladas oticamente. 1 grupo de 8 saídas coletoras abertas isoladas oticamente.

Apêndice A

smar

FSR - Formulário para Solicitação de Revisão

Entrada e Saída Remota Foundation Fieldbus

DADOS GERAIS

Modelo: DC302

Nº de Série: _____

TAG: _____

Utilizando quantos canais?

ENT 1-4 ( ) 9-12 ( )
5-8 ( ) 13-16 ( )

SAI 1-4 ( ) 5-8 ( )

Configuração: PC ( ) Software: _____ Versão: _____

DADOS DA INSTALAÇÂO

Tipo/Modelo/Fabricante do equipamento conectado ao DC302: _____

_____

DADOS DO PROCESSO

Classificação da Área/Risco: ( ) Sim, por favor especifique: _____

( ) Não

Mais detalhes: _____

Tipos de Interferência presente na área: Sem interferência ( ) Temperatura ( ) Vibração ( ) Outras: _____

Temperatura Ambiente: De _____ºC até _____ºC.

DESCRIÇÃO DA OCORRÊNCIA

_____

_____

_____

_____

SUGESTÃO DE SERVIÇO

Ajuste ( ) Limpeza ( ) Manutenção Preventiva ( ) Atualização / Up-grade ( )

Outro: _____

DADOS DO EMITENTE

Empresa: _____

Contato: _____

Identificação: _____

Setor: _____

Telefone: _____ _____ _____ _____ **Ramal:** _____

E-mail: _____ **Data:** _____/ _____/ _____

Verifique os dados para emissão da Nota Fiscal de Retorno no Termo de Garantia anexado ao manual.

Retorno de Materiais

Caso seja necessário retornar o DC302 para avaliação técnica ou manutenção, basta contatar a empresa SRS Comércio e Revisão de Equipamentos Eletrônicos Ltda., autorizada exclusiva da Smar, informando o número de série do equipamento com defeito, enviando-o para a SRS de acordo com o endereço contido no termo de garantia.

Para maior facilidade na análise e solução do problema, o material enviado deve conter, em anexo, a documentação descrevendo detalhes sobre a falha observada no campo e as circunstâncias que a provocaram. Outros dados, como local de instalação, tipo de medida efetuada e condições do processo são importantes para uma avaliação mais rápida e para isto, use o Formulário para Solicitação de Revisão (FSR).

TERMO DE GARANTIA SMAR

1. A SMAR garante os equipamentos de sua fabricação por um período de 18 (dezoito) meses, contados da data da emissão da Nota Fiscal. A garantia independe da data de instalação do produto.

2. Os equipamentos de fabricação SMAR são garantidos contra qualquer defeito proveniente de fabricação, montagem, quer de material quer de mão de obra, desde que a análise técnica tenha revelado a existência de vícios de qualidade passíveis de enquadramento neste termo, comprovados pela análise técnica e dentro dos prazos em garantia. A análise técnica aqui mencionada será realizada exclusivamente pelos laboratórios SMAR, ou efetuados pela empresa SRS Comércio e Revisão de Equipamentos de Automação Ltda., autorizada exclusiva Smar; vide item 4.

3. Excetuam-se os casos comprovados de uso indevido, manuseio inadequado ou falta de manutenção básica conforme indicado nos manuais de instrução dos equipamentos. A SMAR não garante qualquer defeito ou dano provocado por situação sem controle, incluindo, mas não limitado aos seguintes itens: negligência, imprudência ou imperícia do usuário, ações da natureza, guerras ou conturbações civis, acidentes, transporte e embalagem inadequados efetuado pelo cliente, defeitos causados por incêndio, roubo ou extravio, ligação à rede de tensão elétrica ou alimentação imprópria, surtos elétricos, violações, modificações não descritas no manual de instruções, se o número de série estiver alterado ou removido, substituição de peças, ajustes ou consertos efetuados por pessoal não autorizado; instalações e/ou manutenções impróprias realizadas pelo cliente ou por terceiros, utilização e/ ou aplicação incorreta do produto, ocasionando corrosão, riscos ou deformação do produto, danos em partes ou peças, limpeza inadequada com utilização de produtos químicos, solventes e produtos abrasivos não compatíveis com os materiais de construção, influências químicas ou eletrolíticas, partes e peças que se desgastam com o uso regular, utilização do equipamento além dos limites de trabalho (temperatura, umidade entre outros) conforme consta no manual de instruções. Além disso, este termo de garantia exclui despesas com transporte, frete, seguro, constituindo tais ítens, ônus e responsabilidade do cliente.

4. Os serviços técnicos de manutenção em garantia serão efetuados pela empresa SRS Comércio e Revisão de Equipamentos de Automação Ltda, autorizada exclusiva Smar. Os equipamentos com problemas técnicos comprovados deverão ser despachados e entregues no endereço abaixo, com frete pago pelo cliente.

   Dados para emissão da Nota Fiscal de Retorno:
   SRS Comércio e Revisão de Equipamentos de Automação Ltda.
   Rodovia Albano Bachega Km 2,1 – Vicinal Sertãozinho/Dumont Sertãozinho/SP
   Caixa Postal 532 – CEP 14173-020
   IE: 664.156.985-115 CNPJ: 009.005.841/0001-66 Fone: (16) 3513-2500 Fax: (16) 3513-2525
   E-mail: revisoes@srsrevisoes.com.br

5. Nos casos em que houver necessidade de assistência técnica nas instalações do cliente durante o período de garantia, não serão cobradas as horas efetivamente trabalhadas, entretanto, a SMAR será ressarcida das despesas de transporte, alimentação e estadia do técnico atendente, bem como dos custos com desmontagem e montagem quando existirem.

6. O reparo e/ou substituição de peças defeituosas não prorroga sob hipótese alguma o prazo da garantia original, a não ser que essa prorrogação seja concedida e comunicada por escrito pela SMAR.

7. Nenhum Colaborador, Representante ou qualquer outra pessoa tem o direito de conceder em nome da SMAR garantia ou assumir alguma responsabilidade quanto aos produtos SMAR. Se for concedida alguma garantia ou assumida sem o consentimento por escrito da SMAR, esta será declarada antecipadamente como nula.

8. Casos de aquisição de Garantia Estendida devem ser negociados com a SMAR e documentados por ela.

9. O atendimento ao cliente é realizado pela Assistência Técnica SMAR Fone: (16) 3946-3509 (Horário Administrativo) e (16) 3946-3599 (Plantão 24 h) localizado na Matriz em Sertãozinho (SP) ou pelos Grupos de Atendimentos localizados nos escritórios regionais da SMAR.

10. Caso seja necessário retornar o equipamento ou produto para reparo ou análise, basta entrar em contato com a SRS Comércio e Revisão de Equipamentos de Automação Ltda. Vide item 4.

11. Em casos de reparos ou análises deve-se preencher a “Folha de Solicitação de Revisão”, a FSR, contida no manual de instruções, onde deve conter detalhes sobre a falha observada no campo e as circunstâncias da mesma, além de informações sobre o local de instalação e condições do processo. Equipamentos e produtos não cobertos pelas cláusulas de garantia serão objetos de orçamento sujeitos à aprovação do cliente antes da execução do serviço.

12. Nos casos de reparos em garantia, recall ou fora de garantia, o cliente é responsável pelo correto acondicionamento e embalagem e a SMAR não cobrirá qualquer dano causado em transportes.Despesas de serviços ou quaisquer custos relacionados a desinstalação e instalação do produto são de responsabilidade do cliente e a SMAR não assume nenhuma responsabilidade frente ao comprador.

13. **Responsabilidade:** Exceto as condições gerais de garantia para Produtos SMAR, mencionadas anteriormente, a SMAR não assume nenhuma responsabilidade frente ao comprador, e isso sem limitações, quanto a danos, conseqüências, reivindicações de indenização, lucros cessantes, despesas com serviços e outros custos que forem causados pela não observação das instruções de instalação, operação e manutenção contidas em manuais SMAR. Além disso, o comprador também declara inocentar o fornecedor de indenizações por danos (excetuando os custos com consertos ou com a reposição de produtos defeituosos descritos anteriormente), causados direta ou indiretamente por causa de teste, aplicação, operação ou conserto inadequados de produtos SMAR.

14. É responsabilidade do cliente a limpeza e descontaminação do produto e acessórios antes de enviar para reparo e a SMAR e sua autorizada se reserva do direito de não repararem o equipamento nos casos onde assim não for procedido. É responsabilidade de o cliente avisar a SMAR e sua autorizada quando o produto for utilizado em aplicações que contaminam o equipamento com produtos que podem causar danos durante o seu manuseio e reparo. Qualquer dano, conseqüências, reivindicações de indenização, despesas e outros custos que forem causados pela falta de descontaminação serão atribuídos ao cliente. Por gentileza, preencher a Declaração de Descontaminação antes de enviar produtos à Smar ou autorizadas e que pode se acessada em HTTP://www.smar.com/brasil2/suporte.asp e enviar dentro da embalagem.

15. Este termo de garantia é válido apenas quando acompanhado da Nota Fiscal de aquisição.